48355

# Ahoy! COMMODORE USERS

K

Ion International Inc. $2.50/Can. $2.75 May 1984

## THE FUTURE OF COMMODORE?



SPREADSHEETS FOR THE 64

MONITOR YOUR BIORHYTHMS

MEMORY MANAGEMENT

05

0 71896 48355

# You asked for it . . .
# CARDCO has it!

**Another exclusive from Cardco!**

## The LQ/1 (Letter Quality) PRINTER

In response to your demand and need, CARDCO now has available for the **first** time . . . a "Commodore-ready "LETTER QUALITY PRINTER". Just plug it in and print.

Offering standard friction-feed and optional tractor-feed, the CARDCO LQ/1 PRINTER comes complete with built-in interfacing for all Commodore Personal Computers, as well as compatible input for PC, PC jr., TRS-80 and other personal computers.

## "CARDPRINT" C/?+G PRINTER INTERFACE with Graphics

For printers that are not Commodore-ready, Cardco offers the C/?+G PRINTER INTERFACE including all cables and connectors for the following printers: all new Epson MX, RX and FX series, Star Gemini 10X and Delta 10, Prowriter, C.-Itoh 8510, NEC 8023, Okidata 82, 83, 84, 92, 93 and 94, Mannesman Tally Spirit and MT-160, Seikosha GX-100, BMC BX-80 and the Gorilla Banana. Prints the full Commodore character set; graphics, characters, reversed characters and reversed graphics characters.

## "WRITE NOW" WORD PROCESSOR SOFTWARE

An excellent time saver, CARDCO OFFERS THE CØ2 "Write Now" program with built-in 80 column display. You see exactly what will print. All special codes can be transmitted to printers maintaining justification. Easy full-screen editing; works with any printer.

## "MAIL NOW" MAILING LIST SOFTWARE

CARDCO D/01 quickly (in memory) sorts by zip, category, name and state; fully compatible with "WRITE NOW". Other fine features include: User-oriented; menu-driven operation; each disk supports 600 entries. Format can print single, double or triple labels across.

All Cardco Products are available at your local dealers. Write for illustrated literature.




**cardco, inc.**

313 Matheson Wichita, Kansas 67214 (316) 267-6525

**"The world's largest manufacturer of Commodore accessories."**

# Ahoy!

## CONTENTS

## DEPARTMENTS



## FEATURES



## PROGRAMS



*Cover photo © 1984 Ross M. Horowitz*




AHOY! No. 5, May 1984. Published monthly by Ion International Inc., 45 W. 34th St., Suite 407, New York, NY, 10001. Subscription rate: 12 issues for $19.95, 24 issues for $37.95 (Canada $26.95 and $49.95 respectively). Application to mail at 2nd Class postage rates is pending at New York, NY and additional mailing offices.

# VIEW FROM THE BRIDGE

Now we've gone and done it! We boasted in last issue's *View from the Bridge* about how *Ahoy!* has become the most reader-friendly, most visually appealing Commodore magazine going. What we didn't realize was: now we're going to have to live up to that boast every issue! We've managed to do it this time—but can we keep it up? The only way you'll find out is to continue buying us every month (and browsing through the competition on the newsstand).

As for *this* issue:

Who could have predicted anything other than smooth sailing in '84 for the microcomputer company that racked up $1 billion in sales in '83? But in the wake of a top management exodus and plunging stock prices, many Commodore owners are worried about being left high and dry. Don't donate your VIC or 64 to the Smithsonian until you've read *The Future of Commodore?* by Sheldon Leemon. (Turn to page 44.)

Ready to buy your first spreadsheet. . .or even move up to your second? Terry Silveria, co-author of *The Commodore 64 Software Review* among other books on your favorite microcomputer, lays all the information you'll need out in front of you in *Spreadsheets for the C-64.* (Turn to page 18.)

Two demerits for any parent who hasn't been following Richard Herring's *Educational Software* series! In *Part III,* he explains the importance of looking into the ease of operation, manner of error handling, and warranty of your chosen software. (Turn to page 41.)

We've shown you the inside of the 1541 disk drive (February) and the C-64 (March); are you ready for the inside of a BASIC program? Michael Kleinert and David Barron display a little-known side of your terminal input in *That's My Program!?!?!* (Turn to page 61.)

Michael and David have also added *VIC 20 Bug Repellent* to the 64 version introduced last issue. Program bugs will bug VIC users no longer! (Turn to page 37.)

The first program you'll want to test it on is Joe Woyton's *VIC 20 Calculator,* which makes the mathematical prowess of your VIC more accessible than ever before. (Turn to page 17.)

Of course, you'll find plenty of use for your month-old *C-64 Bug Repellent* inside, too. Guaranteed to function as accurately as the finest newspaper horoscope, *I Got Biorhythm* by B.W. Behling will give you all the advance warning you'll need about which days it won't pay to get out of bed. (Turn to page 63.)

On those days, keep your 64 on your night table and play *Air Assault* by Bob Lloret. You're on the receiving end of the assault, catching bombs that plummet toward a helpless (but for you) city. (Turn to page 35.)

If you're up for a slightly more intellectual challenge, Bob's *Math Master* provides skill levels to challenge everyone from elementary schoolers to college profs. (Turn to page 65.)

Want to do more with your VIC or 64? In *Memory Management on the VIC 20 and C-64* (Part I in this issue), Morton (the K) Kevelson furnishes the programs and tutoring you'll need to make 39,423 bytes of memory available for BASIC on the VIC, and 43,007 on the 64. (Turn to page 29.)

When Dale Rupert told us the title of his latest *Rupert Report,* we shivered a little, thinking he was asking for more money. But *Hurray for Arrays!,* far from being a salary demand, is the clearest explanation of arrays and subscripted variables you're likely to find. (Turn to page 67.)

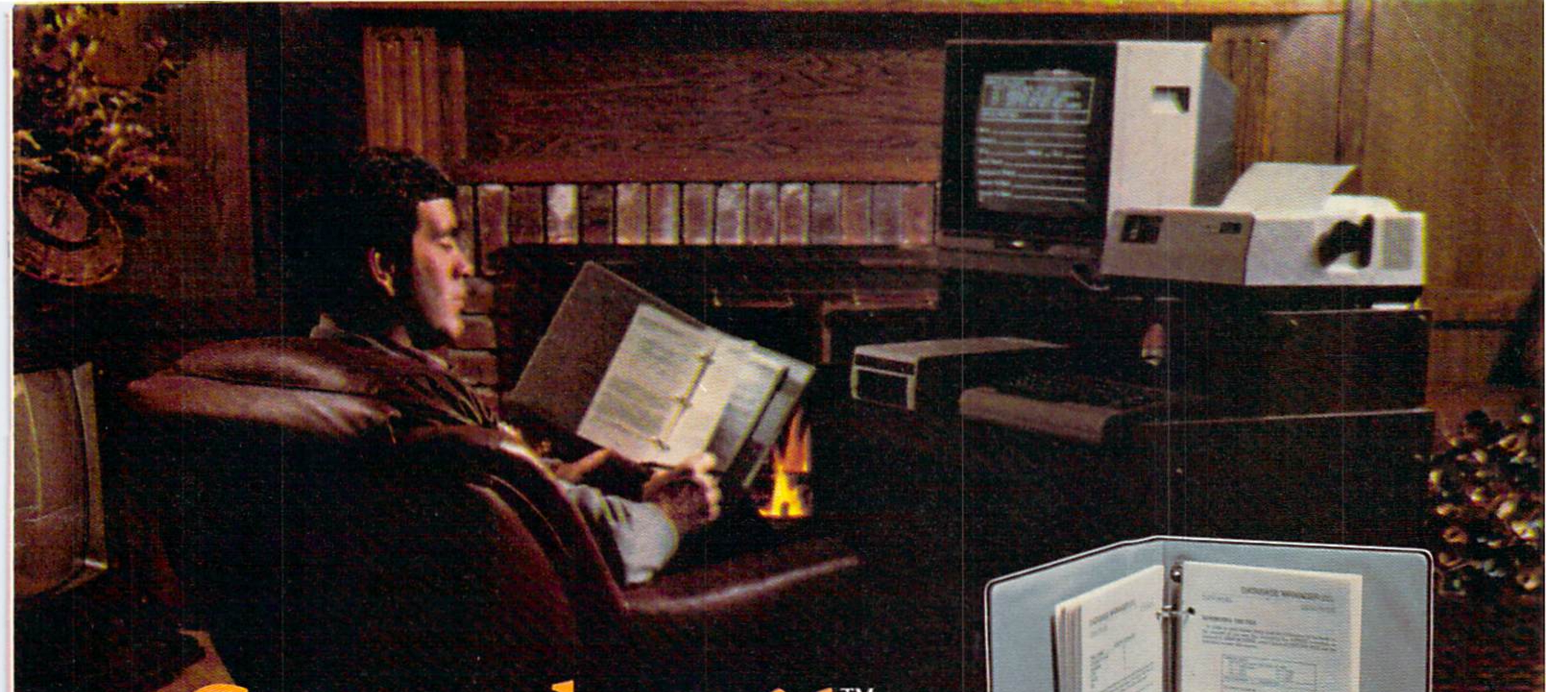
Several of our freelancers who saw the artwork for last month's (and this month's) *Commodares* sitting around the office couldn't figure it out. "What's a big glove got to do with *Commodares*"? they asked. "It's a guantlet, computer geniuses," we replied. "Ever hear of throwing down the gauntlet?" You can pick it up again this issue, as Dale Rupert again attempts to stump our readership. If you meet the challenge, you'll see your name in *Ahoy!* (Turn to page 39.)

Of course, all other regular departments will be found inside. In *Scuttlebutt* we preview the new high-speed disk drives for the C-64 and a host of other hardware, software, and miscellaneous VIC and 64 releases. (Turn to page 8.)

We continue to pack in more *Reviews* than any other Commodore magazine on the stands. (We pack in so much more *news* that it seemed useless to point it out last paragraph!) Included in this issue are *Time Runner, All About Chanukah, The Computer Mechanic,* and *Mastering the SAT!* How's that for variety? (Turn to page 49.)

Say—does anyone read the *Glossary* we run each issue? We've yet to receive our first letter mentioning the whimsical additions we tacked onto several of the entries while Mike Schneider wasn't looking! Of course, letters on that or any subject at all would be welcome for *Flotsam,* our favorite section of *Ahoy!* Because we don't have to write it? No—because it's the one place where we and our readers can really "interact!" We can't chart our course for future issues without hearing from the captains of the ship: each and every reader of *Ahoy!* So enjoy this issue—but not so much that you don't have time to write us!

***—Steve Springer***





# Commodore 64™ Owners, Relax...

## with Mirage Concepts software

Mirage Concepts has mastered the art of uncomplicating software. Before you buy—we help you determine which Mirage Concepts package will meet your need. No guesswork! With your purchase comes a menu-driven program ranked by independent evaluators nationwide as among the finest available. Relax as you learn how to operate your program with clear, concise tutorials written by professional writers... not programmers. For consultation on your special questions, technical support personnel are standing by on a toll-free basis.

*For Brochures, Support and Information, Call...*
(800) 641-1441
*In California, Call...*
(800) 641-1442

**DATABASE MANAGER, $89.95**
• 100% Machine Language • Free Form Design • Sort On Any Field • Calculated Fields
• Interfaces to W.P. • Record Size = 2,000 Characters

**ADVANCED REPORT GENERATOR, $49.95**
• Companion to Database • Totals and Subtotals • Field Matching • Expanded Reports
• Sorting (Up & Down) • Calculated Fields

**WORD PROCESSOR,** Professional Version $89.95
• 80 Col w/o Addt'l Hdwr • 100% Machine Language • Spelling Checker (30,000 Words)
• Over 70 Single Keystroke Commands • Printer Command File • Interfaces to Database

**WORD PROCESSOR,** Personal Version $39.95
• 100% Machine Language • True Word Wrap • Printed page/line/character counters
• Right Justify, Center • Printer Command File • Interfaces to Database



**MIRAGE CONCEPTS, INC.**

2519 W. Shaw Ave., #106 • Fresno, CA 93711

TM—Commodore 64 is a Registered Trade Mark of Commodore Electronics, Ltd.

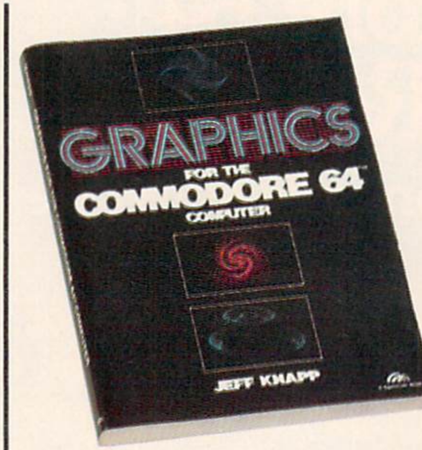
Reader Service No. 212

# Prentice-Hall speaks a language other publishers have forgotten.

# English.*



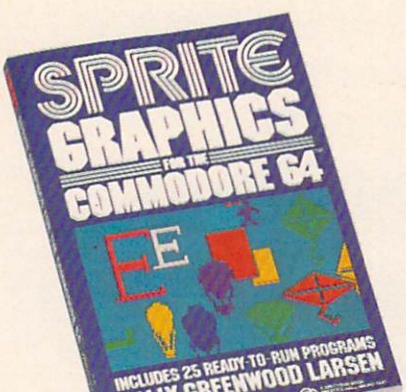

**PROGRAMMING YOUR COMMODORE 64 IN BASIC** by Mario J. Eisenbacher. An easy-to-digest intro that includes, at each level, new vocabulary, lively sample programs, and exercises/answers. Two entire chapters on graphics. $12.95



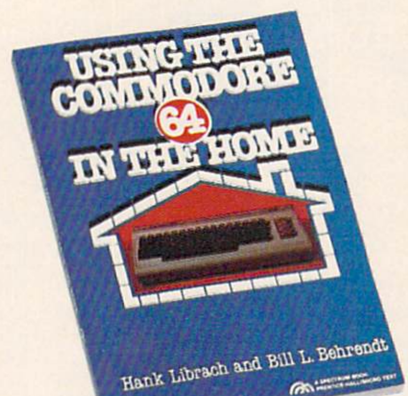

**GRAPHICS FOR THE COMMODORE 64 COMPUTER** by Jeff Knapp. This collection of easy-to-master programming techniques and BASIC programs unleashes amazing graphics. $12.95



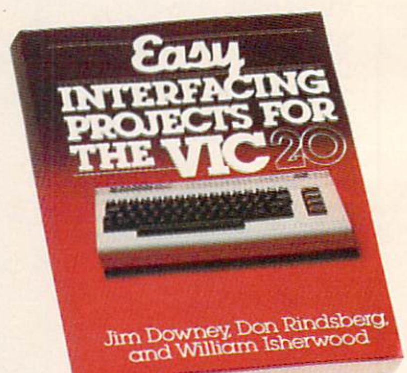

**MUSIC AND SOUND FOR THE COMMODORE 64** by Bill L. Behrendt. How to use the Commodore 64's Sound Interface Device and how to write programs that match the sounds of various hand instruments. $14.95




**SPRITE GRAPHICS FOR THE COMMODORE 64** by Sally Greenwood Larsen. Shows how to produce high resolution, color, animated graphics. Twenty-five read-to-run sample programs. $15.95




**USING THE COMMODORE 64 IN THE HOME** by Hank Librach and William Behrendt. Home of the future! Twenty original programs for check book keeping, loan payments, family nutrition, education, games, and much more. Book/disk available. $10.95




**EASY INTERFACING PROJECTS FOR THE VIC-20** by James Downey, Don Rindsberg, and William Isherwood. Dozens of interfacing projects written in BASIC and specifically designed to maximize the VIC-20's power. $12.95

## PRENTICE-HALL/THE LEADER IN COMPUTER PUBLISHING

For more information about our computer books and software, write to us at the address below. Dealer inquiries welcome.
Prentice-Hall, General Publishing Division, Englewood Cliffs, N.J. 07632

*We guarantee that all our guides are easy to read and simple to apply without the aid of a reference library, a computer salesperson, or a niece who just graduated from M.I.T.

**Reader Service No. 224**

# FLOTSAM

Regarding the *Rupert Report* in your premier issue, there is one more way to control the cursor: access the KERNAL routine, PLOT, from BASIC.

The program on page 89 can be rewritten as follows: delete lines 4 through 100 and substitute these two lines:

```
1020 POKE781,ROW:POKE782,COL:POKE
783,0
1030 SYS65520
```

Here is how it works. Locations (decimal) 780 to 783 shadow four of the processor registers:

| Decimal Address | Register |
|---|---|
| 780 | .A Accumulator |
| 781 | .X X register |
| 782 | .Y Y register |
| 783 | .S Processor status |

In other words, when SYS is executed, the contents of these four locations are loaded into their respective processor registers, then execution of the machine language routine begins. Upon completion of the ML routine, the contents of the four registers are stored to their respective shadow locations just before control is returned to the BASIC program. The two lines I have provided load the X and Y registers, clear the carry flag, and call PLOT saving considerable time, memory, and programming.

Hanson Kappelman
Pittsburgh, PA

On a recent trip to the big city, I was pleased to find *Ahoy!* #1. I have a Commodore 64, and would like to find a program on genealogy, to construct a family tree. Please let me know if such a program exists for the 64, and where I could get it.

William C. Brauch
Canaan, VT

*We aren't aware of any such program, but if any of our readers are, please let us know.*

Right off the bat, I want to complement you on one fine publication. I am looking forward to many hours of study based on the February '84 issue alone. I will be subscribing shortly, but I do want to observe for a few months. I have already introduced your publication to the users group in this area, and I plan on doing the same with mine.

Dean G. Thomas
Edinburg, PA

The New York City VIC 20/C-64 User Group (NYCUG) was organized slightly over a year ago with fewer than ten members. Today, we've grown into a group of approximately fifty members. We presently have a local BBS called VTUG and are setting up a large CP/M public domain library of over 100 disks. Will you please list our user group in your publication? Address is 436 E. 69th Streeet, New York, NY 10021; phone number is (212)787-2854 (VTUG BBS (212)534-3149).

Jocelyn Sue Woods
New York, NY

*This seems as good a place as any to extend an open invitation to users groups across the United States and Canada (and elsewhere) to write us with their particulars: location, meeting time, dues, etc. We'll publicize as many as we can, either in* Flotsam *or in a separate section devoted to the subject.*

---

***CORRECTION***

A printing error creeped into March's *Rupert Report (The City of Truth)*. The two one-line computer runs on page 17 should have read PRINT A=5 and PRINT A=0. We apologize for any confusion this might have caused.

---

# SCUTTLEBUTT

***HIGH-SPEED DISK DRIVES FOR THE 64 • STRESS REDUCTION SYSTEM • GRAPHICS PROGRAMS • @*&!!%# COMPUTERS! • WORD PROCESSING, SPREADSHEET, AND DATABASE SOFTWARE • MUSIC SYNTHESIS PROGRAM • INTERFERENCE REJECTION CABLE • NEW COMMODORE BOOKS • VIC GAME RELEASES FROM IMAGIC AND PARKER BROTHERS***

## *THE SPOILERS*

How easily we get spoiled. When we first bought our VIC-1541 or Commodore 1541 disk drives, we were ecstatic over the improved program access, faster and simpler than with datasette storage. But for many of us, even the relatively brief waiting periods necessitated by the disk drives have become too long.

At least two drives now on the market, and a third on its way, will cut that waiting time substantially. Even now, Morton K. is ripping their guts out: he'll present a detailed analysis in an upcoming issue.

For now, here's a brief look at the high-speed disk drives in question.

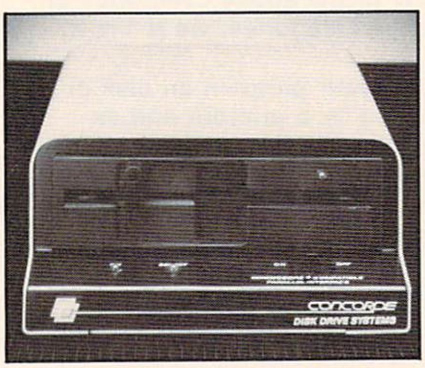
Concorde's C-321P system, featuring a high-speed parallel interface, purports to be twice as fast as the 1541 in storing and retrieving information. 100% compatible with the C-64 and its software, the unit features built-in glass bonded heads, a brushless D.C. motor, and an opto-isolator LED combination to provide instantaneous correction for drive speed variations and to maximize track zero establishment. It uses full ball bearing mountings, with no pulleys or belt, to minimize spindle runout and to provide a 10,000 hour mean time between failure rate. Suggested price is $389.

The MSD Super Disk Drive (SD-1) from Micro Systems Development, Inc., makes many of the same claims, including that of complete compatibility with the entire Commodore line and doubling the operating speed of the 1541. It features a 4K buffer memory, serial and IEEE parallel interface, and an 18-second disk formatting speed. The SD-2 Dual Disk Drive, configurable as two drives or one, has a 6K buffer memory. Suggested price for the SD-1 is $399; for the SD-2, $695.

Commodore's own SFS 481 Fast Disk Drive is not yet available. As with most of Commodore's recently announced new product line, its release date is indefinite. But the unit is expected to deliver a maximum data transfer rate of 1675 bytes per second, with an average access time of 295 milliseconds.

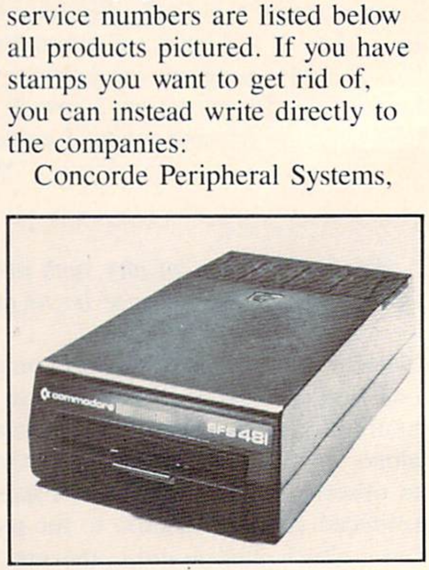
More detailed information will follow. In the meantime, you can obtain literature on these high-speed drives (and any other product pictured in *Scuttlebutt*) by circling the appropriate number on the reader service card located between pages 94 and 95. Reader service numbers are listed below all products pictured. If you have stamps you want to get rid of, you can instead write directly to the companies:

Concorde Peripheral Systems,





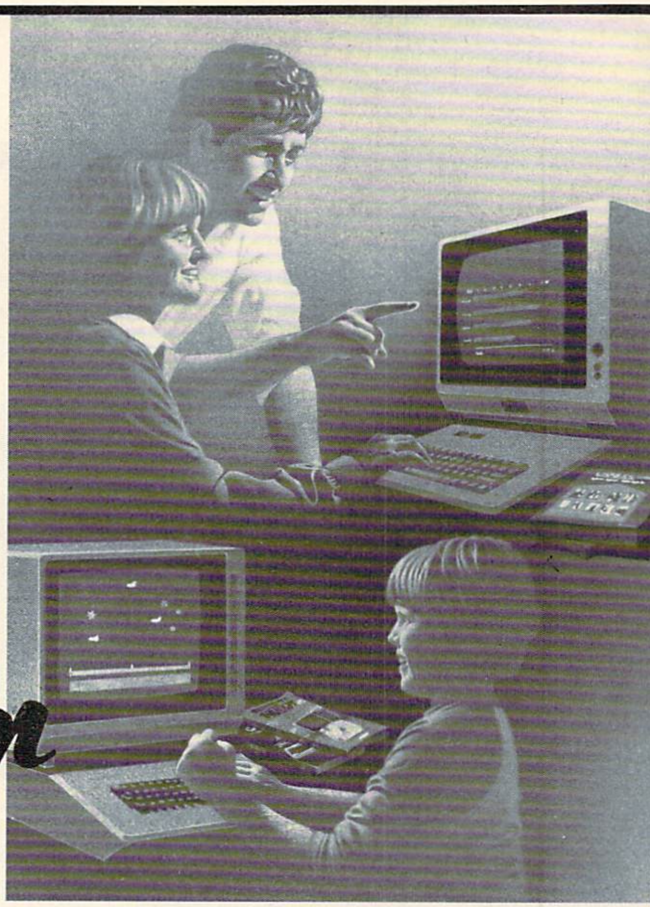





*For the fastest drive of your life, try Concorde's C321P, MSD's Super Disk Drive, or Commodore's SFS481.*
*READER SERVICE NO. 176* *READER SERVICE NO. 177* *READER SERVICE NO. 178*

# Hayden turns an ordinary day into...

# *Great Family Fun*



**Introducing Hayden's new line of top-notch aids for Commodore users—a complete array of books, software, and videotapes that gives you the right amount of "over the shoulder" guidance you need.**

**Problem Solving in BASIC with the Commodore 64™**
(Belove-Laiserin) Don't frustrate yourself trying to understand technical instruction manuals. Sit back. Call in the rest of the family. Turn on your set. Load the cassette. And let your VCR do the rest. What you get is a 30-minute visual tour on programming—how to state a problem, how to break it into the workable modules or subroutines, and then how to actually write BASIC instructions for each step.
**Available in both VHS (#6459) and BETA (#6460) format, $59.95 ea.**

## Everybody loves a good challenge...

**Lost in Space: Programming Adventures in BASIC I**
Test your ingenuity. Help Dr. W and his colleagues escape from a derelict spaceship. How? Through the ship's computer, of course. So much fun you won't realize that you're actually learning how to program in BASIC.
**Commodore 64 version, #7635, $24.95**

## And for the serious programmer in the family...

**Commodore 64™ Assembly Language Programming**
**VIC™ Assembly Language Programming**
(Derek Bush & Peter Holmes) Great new assembly language guides that get you up to speed in practically no time at all. Easy to use. Avoids tedious math conversions. Gets you into simple programming right away. Leaves you ready to start writing your own assembly programs—even the fast-moving arcade type.
**Commodore 64 (#7620), VIC (#7520), $29.95 ea.**

HAYDEN

**Order by Phone 1-800-631-0856**
operator AY54 • In NJ call (201) 393-6315

**Mail to:** Dept. AY54 • Hayden Book Company
10 Mulholland Drive • Hasbrouck Heights, NJ 07604

Please send me the book(s) indicated below by code number. If I am not completely satisfied I may return the book(s) undamaged, within 10 days for a complete refund. I am enclosing $2.00 to cover postage and handling.
☐ Enclosed is my check or money order. Bill my ☐ Visa ☐ MasterCard

| | | | |
|---|---|---|---|
| | | | |

Name ____________________

Address ____________________

City ____________ State ________ Zip ________

Visa/MasterCard # ____________ Exp ____________

Signature ____________________

Residents of NJ and CA must add sales tax. Prices subject to change.



Reader Service No. 208

Inc., 23152 Verdugo Drive, Laguna Hills, CA 92653.

Micro Systems Development, Inc., 10031 Monroe Drive, Suite 206, Dallas, TX 75229.

Commodore Business Machines, Inc., 1200 Wilson Drive, West Chester, PA 19380.

## *SILICON ALLEY*

Rags to riches story are nowhere as prevalent as in the music industry. So the rise of Entech's *Studio 64* between the time we reported on it in *Scuttlebut* (March) and now should not be surprising. At the Winter Consumer Electronics Show, Kapri International Distributors honored the music word processor with a Gold Disk Award for excellence in educational programming. And now a songwriting contest centered around the program has been announced by Entech.

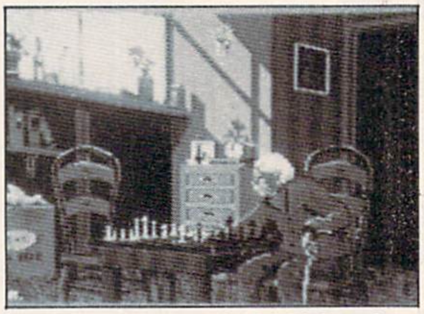
The composer of the best song written on *Studio 64* receives $1000 and recording time at a Hollywood studio, with musicians, an arranger, and a producer available to turn his ditty into a chart-topper. Second prize is $500 and recording time; third, recording time alone.

Submissions are due November 1; winners will be chosen December 15. For complete rules and entry blanks write Computer Song Writing Contest, P.O. Box 881, Sun Valley, CA 91353.

## *KUDOS FROM KAPRI*

If you're dying to know the other winners of Kapri's Gold Disk awards, they were: *Disk Manipulator* (Entech) for best utility programming; *Temple of Apshai* (Epyx) for best recreational software; *Paperclip* (Batteries Included) for best word processor; *Multiplan* (HESWARE) for best business program; and *Complete Personal Accountant* (Futurehouse) for best home program. (Good think Michael Jackson doesn't manufacture software.)

Kapri International, 11671 Sheldon St., Bldg. K, Sun Valley, CA 91352.



**Paintpic** ***boasts of 65,000 brushes.***
***READER SERVICE NO. 180***

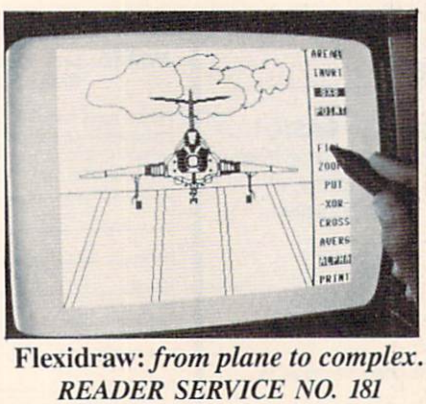
## *GRAPHIC DISPLAYS*

No edition of *Scuttlebutt* would be complete without a look at a couple of good graphics programs.

*Paintpic* from Kiwisoft, a multicolor bitmap drawing, painting, and drafting program, is keyboard- or joystick-controlled. It features automatic shapes, tilt, filled figures, perspective, 65,000 brushes and eight brush widths, and mirroring, rotating and halving capabilities.

The program utilizes all keyboard characters and symbols, and all 16 colors of the C-64. Also included is a BASIC program for putting your pictures into your own games and programs. $39.95 for disk, $35 for tape.

Kiwisoft Programs, 18003 L Sky Park South, Irvine, CA 92714.



**Flexidraw:** ***from plane to complex.***
***READER SERVICE NO. 181***

*Flexidraw* from Inkwell combines graphics software with a light pen to let the user produce anything from simple freehand sketches to complex CAD-type drawings. Menu consists of such automatic graphic selections as instant line, box, circle, and rubber band choices, zoom for detailed work, two separate work areas, Put/Get commands for manipulating and transferring images, and shading and pattern fills.

Inkwell Systems, 7760 Vickers Street, P.O. Box 85152 MB290, San Diego, CA 92138.

## *RUN INTERFERENCE*

The Video*Clear interference rejection cable will reduce radio frequency interference in TV's being used as computer monitors. The cable is easily installed, with all adapters for interfacing included and no modifications to computer or TV necessary.

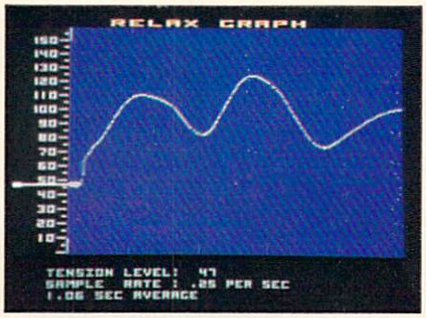
For both the 64 and VIC, the cable retails for $14.95 plus $2.00 shipping and handling from Computer Associates, Inc., Box 683, West Fargo, ND 58078.

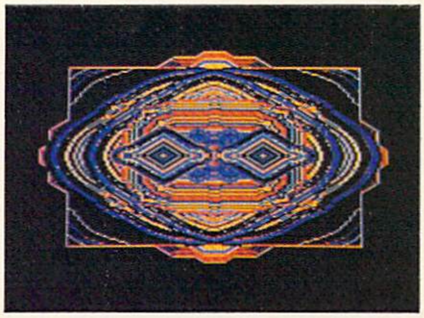
## *GOTO @*&!!%#*

Ever jiggle your computer just right and lose an eight-page letter in memory? Leave a disk on a hot radiator overnight? Record the latest Devo album on the cassette that had held your database? *@*&!!%# Computers!*, a new newsletter, offers you a chance not only to vent your frustration, but to turn it into profit.

Send your tale of programming woe *@*&!!%#* (or *Expletive*) *Computers* at P.O. Box 553, Mt. Freedom, NJ 07970. If it is accepted, you'll receive $25.00. If you'd rather seek solace in the miseries of others, a one-year subscription (10 issues) is $6.00.





*Even in today's high-tech world, the* **Relax** *hardware/software package can help you to mellow out—or program your brain with subliminal messages. At bottom are two graphics representations of stress levels.*
*READER SERVICE NO. 148*

## *SYNAPSE: THE END OF NERVES*

No one is safe from stress anymore—not in 1984, with the fast lane running through every town in America. To that end comes *Relax,* a hardware/software package from Synapse. The menu-driven program combines biofeedback, autogenic training, and progressive relaxation techniques to turn your C-64 into a stress reduction system.

As a sensor headband monitors muscle tension and an audio tape guides you into deep relaxation, you'll actually see your stress level on the monitor. The four displays include a graph that assigns a numeric value to your stress level, a fluctuating geometric pattern that graphically represents your stress, a face that tenses up or relaxes as your face does the same, and a balloon floating across the countryside. A control unit lets you set your own relative stress level, and dump it to a printer.

*Relax* can also be preprogrammed with a subliminal message to help the user in such areas as losing weight or quitting smoking. If testee Bob Sodaro hadn't needed *Relax* prior to hearing that, he certainly needed it afterward. But Roxana Wolosenko, Product Manager for Synapse, assured Bob that he was not in the grips of an Orwellian mind control device. "A 17-year-old can't hook his girlfriend up to it and have it say, 'Trust me,'" Roxana stated. "A subliminal message can't make you do something you didn't want to do in the first place. And the program protects against the user getting a message he doesn't want by flashing the message on the screen."

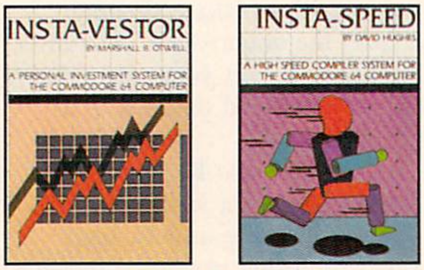
Also of use is the *Relax* workbook, which helps you to understand and deal with your reactions to stress, and create a personal stress profile. The entire package will retail for around $99.95.

Synapse, 5221 Central Avenue, Richmond, CA 94804.

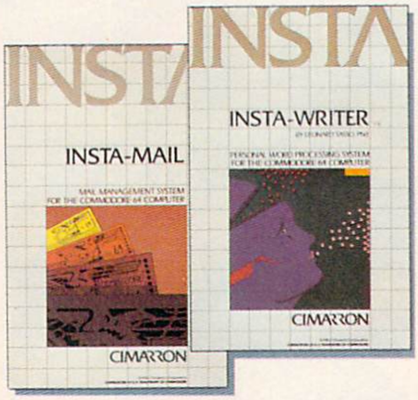
## *TASTES LIKE FRESH POKED*

InstaSoftware is the new name for what had been Cimarron Corporation, recently acquired by Microsci Marketing, a division of Standun Controls. Corporate enough for you? Then let's run down the Insta Series of business-inspired utilities for family use.

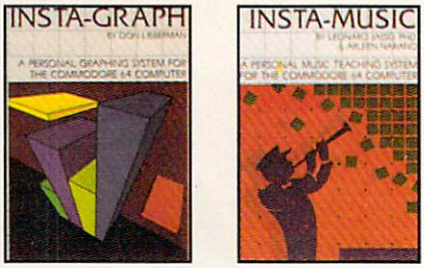
*Insta-Writer* and *Insta-Mail* are designed to work together. The former is a cartridge-based word processor designed for ease of use. The latter is a machine-code module that interfaces with *Insta-Writer* to process names and ad-



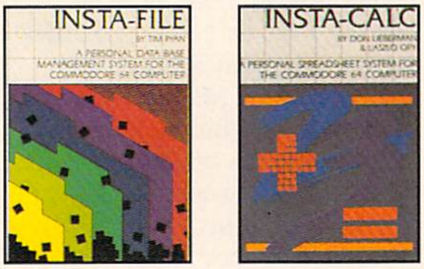

*READER SERVICE NOS. 153, 154*



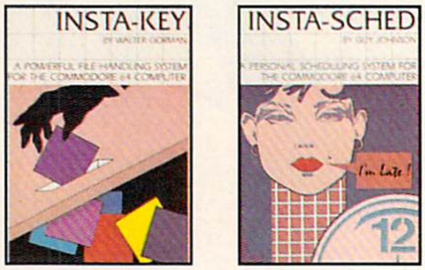

*Insta Software markets a line of home software inspired by business applications.* **Insta-Mail** *and* **Insta-Writer,** *a name and address file and word processor, work together.*
*READER SERVICE NOS. 157, 158*




*READER SERVICE NOS. 159, 160*




*READER SERVICE NOS. 155, 156*




*READER SERVICE NOS. 161, 162*

dresses as listings or labels (99 names per file, unlimited files), and automaticaly merge or insert into *Insta-Writer* documents.

*Insta-Vestor* lets you enter stock names and histories, then calculates, defines, compares, and optionally graphs stock movement characteristics.

*Insta-Speed* is a compiler that converts your program into a more efficient, faster-running form.

Others include *Insta-File* (database manager for non-programmers), *Insta-Calc* (entry-level spreadsheet), *Insta-Graph* (create, edit, and combine multicolor bar and dot graphs), *Insta-Music* (write, play, and store compositions), *Insta-Key* (keyed file access method), and *Insta-Sched* (machine code time scheduling program).

Insta Software, 2158 Hathaway Street, Santa Ana, CA 92705.

## *THE WHOLE SPECTRUM*

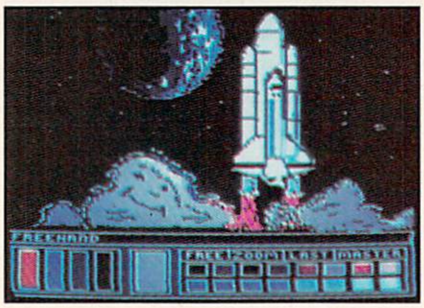
Cave walls, stone tablets, and papyrus all took their turns through the ages as receptacles for artistic inspiration. Canvas and paper are not passé yet but Suncom's *Animation Station* is yet another computer-driven nail in their coffin.

One of the tablet's many features is a 3 X 4 surface that matches the configuration of a TV screen or monitor, allowing the graphics drawn to fit within the borders of the screen. The *Station* is bundled with its own graphics utility program, entitled *Doodler.* Coming software titles include *Pixel Blaster* (an educational program), *Floor Planner* (room layout and design), *Para Graph* (word processor), *Abacus* (math education), *Animator* (just for fun), and several others.

Suncom, 650 Anthony Trail, Suite E, Northbrook, IL 60062.



**Animation Station** *lets you draw on the 64's graphics capabilities.*

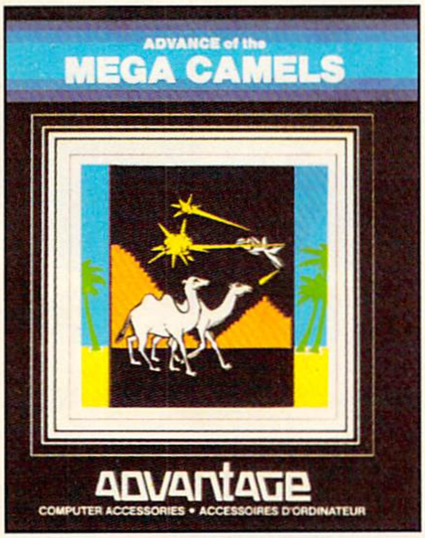



*You too can see your talent rocket.*
*READER SERVICE NO. 163*

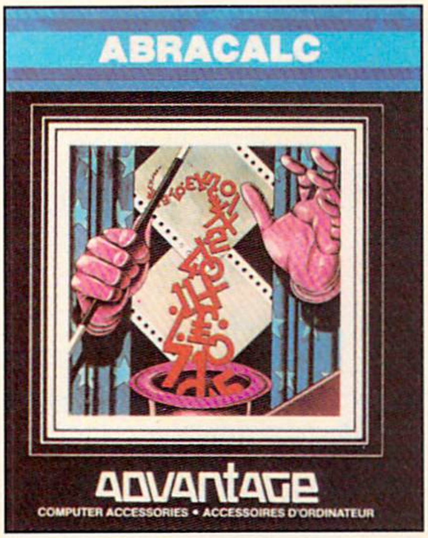
## *TEXT, MASTER*

The Canadian firm Advantage Software, who previously brought us such playful entries as *Advance of the Mega Camels,* are getting down to serious business with their two latest releases for the 64 and VIC.

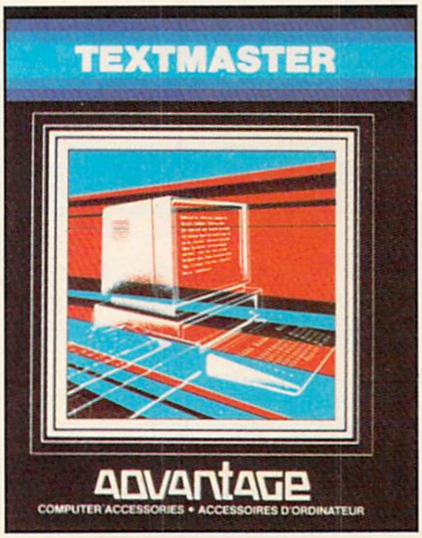
*Abracalc* ($79.95 Canadian) is a spreadsheet program (64 only) in cassette or disk that allows the user to set up home budgets and make business calculations, cash flow analysis, sales forecasts and more, all responsive to what if?




*READER SERVICE NO. 164*




*READER SERVICE NO. 165*



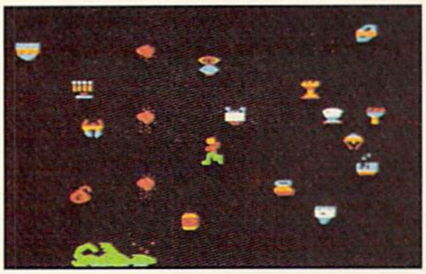

*READER SERVICE NO. 166*

*The three "f's"—fun, finance, and phraseology—are covered by the Canadian-based Advantage Software.*

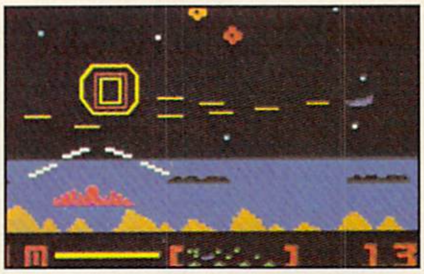
input from the user.

*Textmaster* ($39.95 Canadian) is a word processor for the 64 and "all memory configurations of the VIC 20" that operates on the "two page" principle, which separates the text editing and text processing functions. Other features include French or English screen language; selectable border, screen and character colors; automatic word wrap; page hunt function to locate a word or phrase; pagination, and all major word processing functions including file linking for documents of unlimited length.

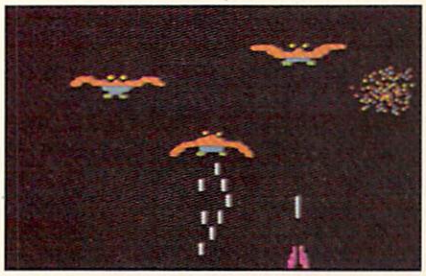
Advantage, 1020 Meyerside Dr., Unit 8, Mississauga, Ontario L5T 1K7.



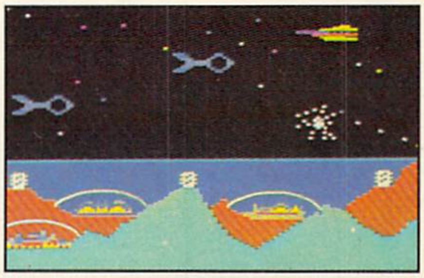
**Dragonfire:** *slay a fire-breather.*
*READER SERVICE NO. 167*



**Nova Blast:** *defend four cities.*
*READER SERVICE NO. 168*



**Demon Attack:** *longtime bestseller.*
*READER SERVICE NO. 169*



**Atlantis:** *shoot enemy ships above.*
*READER SERVICE NO. 170.*

## *LET THE GAMES BEGIN*

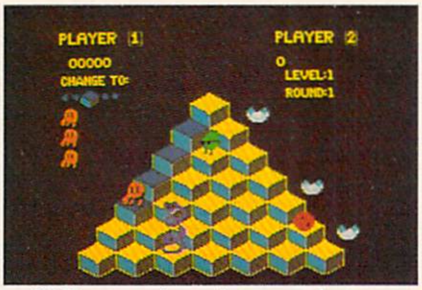
Owners of the VIC 20 with a lust for leisure should sit up and take notice: two major software companies are throwing major releases your way.

Imagic will soon be making these offerings:

*Dragonfire* is a free-form quest scramble across multiple screens. As a young prince the player reclaims his/her kingdom from a fire-spitting dragon. *Nova Blast* is a streak-and-shoot above a waterlogged planet. Defend four domed cities from alien scum! *Demon Attack* is a slide-and-shoot featuring enemies that divide and try to conquer you. A best-seller in years gone by. In *Atlantis,* the player has three gun emplacements and must blast ever-accelerating, ever-descending enemy ships above.

From Parker Brothers, VIC 20 owners can look forward to *Tutankham* (maze mummery), *Q*Bert* (a color-changing, hopping whatsit) and *Frogger* (64 versions reviewed in this issue.)

For the 64, Parker Brothers will soon be releasing *Q*Bert,*

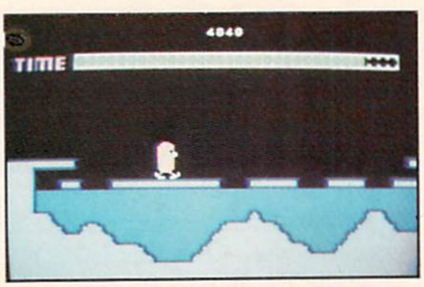



***Q*Bert: a hopping good time.***
**READER SERVICE NO. 171**

*Gyruss* (a spiraling shoot-out in space), *Star Wars* (featuring vector like graphics like the forceful arcade hit), *Popeye* (fisticuffs florentine on multiple levels), and *James Bond* (the long-awaited multiple screen run-and-sleuth).

Deep in the frozen north lives a little-known race of beings known as sno-birds. One of these tiny creatures is the star of *Snokie*




**READER SERVICE NO. 172**

from FunSoft. The title character has been separated from his girlfriend Cara by the evil grodies. The lovesick sno-bird must traverse fifteen different screens to rescue his quail.

*Snokie* features a "just for fun" mode, seven time-controlled game phases, and the ability to retain high scores.

FunSoft, 28611 Canwood, Agoura, CA 91301.

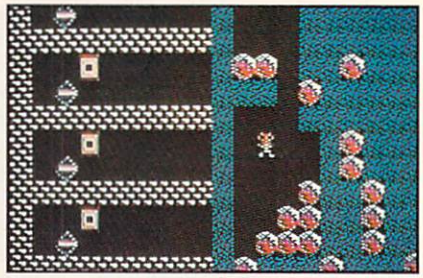
Two new 64 adaptations from Broderbund Software are designed to tax your mind as well as your joystick hand.

Reminiscent of TV's *Concentration, Matchboxes* throws a grid of 36 numbered boxes up on the screen. You must look under the boxes, each of which hides a creature, character, or object of some kind, and match identical squares. Additional modes require players to solve word puzzles, with frontward, backward, and scrambled variations. You can also design your own word puzzles. On disk.

*Operation Whirlwind* plops you into frontline combat, where you must develop and implement strategy that will achieve the most decisive victory (there are five varieties: Questionable, Marginal, Tactical, Strategic, and a Breakthrough). A typical game takes between one and three hours to complete. A colorful battle map and manual are included. On disk.

*Boulder Dash* from First Star sends Rockford on a quest for precious jewels through sixteen mystical caves and five difficulty levels. As boulders crash and fireflies swirl around him, he tries to collect enough diamonds for his escape tunnel to be revealed. A growing amoeba and an enchanted wall add to the challenge.

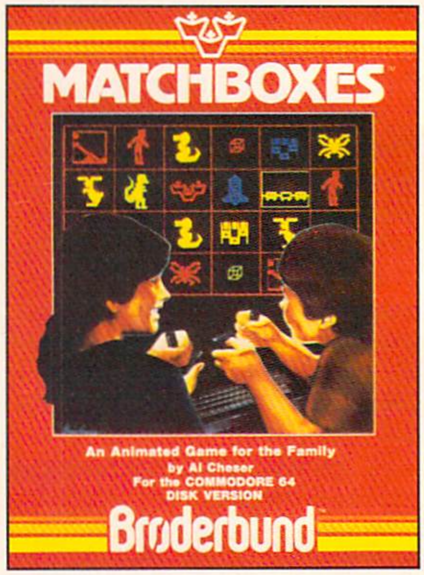
First Star Software Inc., 22



**Boulder Dash** ***from First Star.***
**READER SERVICE NO. 173**

East 41st Street, New York, NY 10017.

Why battle aliens on your C-64 when you can mix it up with such tangible enemies as tooth decay? *Plaqueman* from HomeComputer Software thrusts you into a maze full of teeth, rotting away under the touch of the Pla-




***Safe for children to play with.***
**READER SERVICE NO. 174**

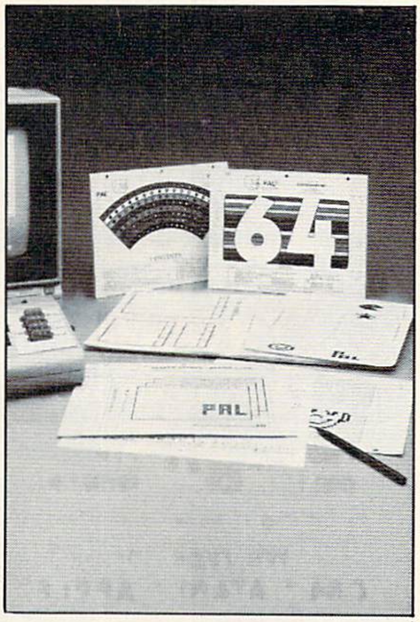
que. Each time they pass over a tooth, they leave behind another layer of slime, eventually turning the tooth black. Plaqueman passes over the same teeth, cleaning them or drilling and filling them where necessary. The Plaque pursue our hero, too. Four tubes of fluoride toothpaste located in the bottom corners of the maze can make him immune to their touch.

HomeComputer Software, Inc., 1307 S. Mary/Suite 209, Sunnyvale, CA 94087.

### *LET THE SLEEPERS AWAKE*

An addition to the Jance home security system mentioned in the March *Scuttlebutt*: a software subroutine that will dial a number (police, neighbor, etc.) and sound an alarm over the phone now comes as standard equipment on both the wireless and hard wire units. Up to 100 numbers can be entered into the system: enough to wake up an entire small town or apartment building.

Jance Associates, P.O. Box 234, East Texas, PA 18046.




*READER SERVICE NO. 202*

### *YOU'VE GOT A PAL*

Programmer's Aids & Logs (PAL to its friends) is a collection of worksheets designed to make program writing easier. Included are cut-out aids, specially designed grid sheets, and programming forms, as well as a full-color chart and guides to POKE's, character strings, sprites, and more. Price is $9.95, plus $2.00 postage/handling.

P.M. Products, 4455 Torrance Blvd., # 177, Torrance, CA 90503.

### *HERE'S A SWITCH*

New from Bytes & Pieces: a reset switch for the Commodore 64, enabling you to recover control without turning off the computer when a program gets hung up in BASIC.

The switch attaches with two solder connections. It can be either externally mounted in a separate box, or mounted through a hole drilled in your computer cover.

Price is $9.95, plus $2.00 postage.

Bytes & Pieces, 550 N. 68th St., Wauwasota, WI 53213.

### *MATH PROGRAMS*

A series of six math programs has been released by Quality Input, Inc. Packages include Basic Mathematics, Basic Algebra, Intermediate Algebra, Geometry, Problem Solving, and Consumer Math, each containing up to 10 programs and oodles of practice questions.

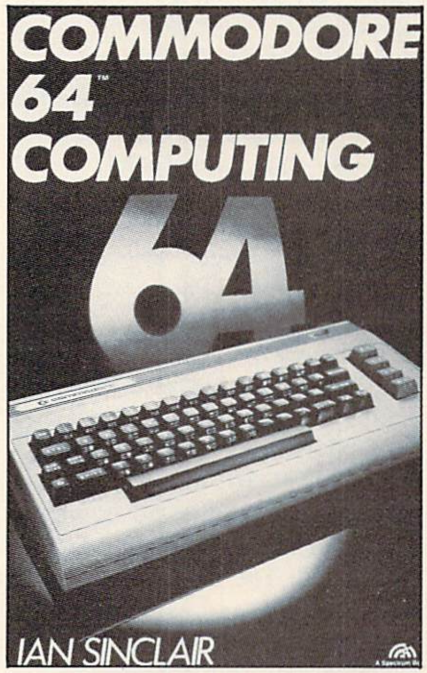
Designed for home or classroom use, the programs let the user print out the number correct, number attempted, and percentage correct at the end of each section.

Quality Input, Inc., 309 W. Beaufort, Normal, IL 61761.

### *HARDWARE AND SOFTWARE HARDCOVERS AND SOFTCOVERS*

A longish subtitle, true, but so is the roster of recent book releases on the C-64 and VIC 20 computers. Here, listed by publisher, is a sampling of titles.

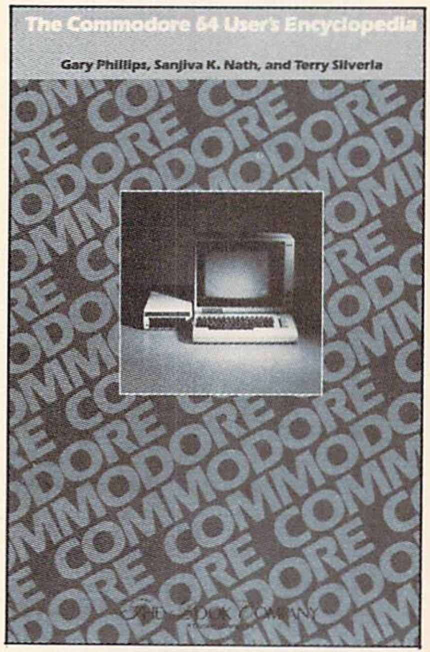
From Melbourne House Software, Inc., Dept. CS, 347 Reedwood Drive, Nashville, TN 37217




*READER SERVICE NO. 195*

(Phone 615-361-3738):

*Commodore 64 Exposed* begins with a step-by-step explanation of BASIC and goes on to explain each of the Commodore BASIC commands in an alphabetical format. Written with the novice in




*READER SERVICE NO. 196*

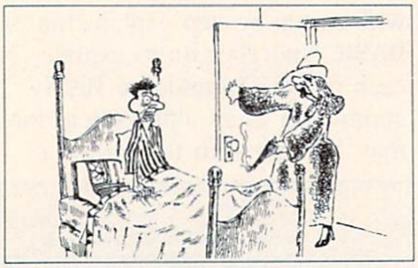
mind. $14.95; add $2.00 for mail orders.

*VIC 20 Exposed*, also for the beginner and also $14.95, includes full-page diagrams illustrating the memory map, chip layout, and other facets of your VIC.

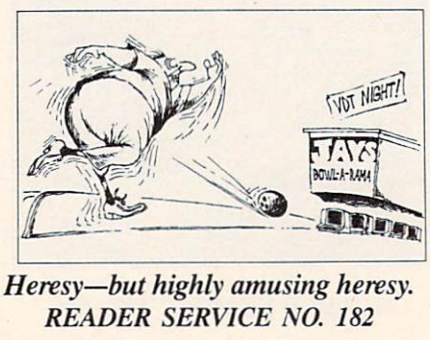
From the Hayden Book Company, 50 Essex Street, Rochelle Park, NJ 07662 (Phone 201-843-0550):

*Commodore 64 Programs for the Home* contains over 40 programs designed for the first-time user. $14.95.

*I Speak BASIC to my Commodore 64* is a field-tested literacy course to introduce the reader to 64 BASIC. The book offers learning objectives and explanations of key terms. It is designed as a teaching aid, with student text selling for $9.75, teacher's $18.75, exam set $15.00 and classroom set $200.00.



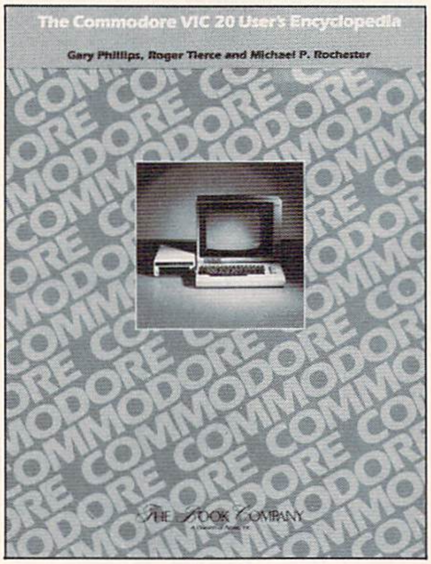


***Heresy—but highly amusing heresy.***
***READER SERVICE NO. 182***

*The Unofficial I Hate Computers Book* places bellylaughs at a high premium—$4.95 for 76 full-page cartoons. But how can you put a price tag on a bellylaugh? A few of our favorites are reproduced here.

From the Book Company, 11223 South Hindry Avenue, Los Angeles, CA 90045 (Phone 213-410-9466):




***Helpful reference for VIC users.***
***READER SERVICE NO. 183***

*The Commodore VIC 20 User's Encyclopedia* and *The Commodore 64 User's Encyclopedia* provide the beginner/intermediate with a convenient reference for some of the problems and decisions encountered in computing. $14.95 each.

From Prentice Hall, Englewood Cliffs, NJ 07632 (Phone 201-592-2640):

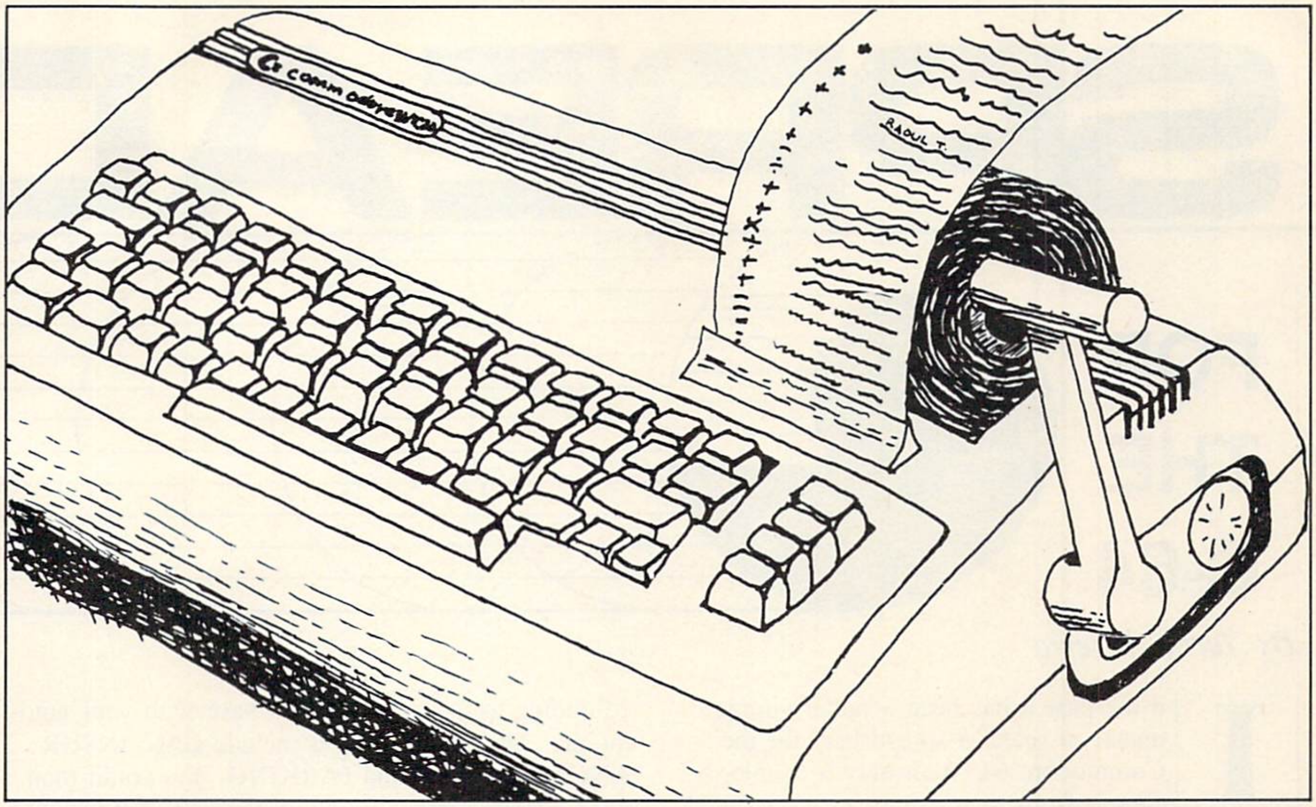
*Commodore 64 Computing* functions as a reference book and how-to guide. $12.95 paper, $19.95 cloth.

*Using the Commodore 64 in the Home* provides 20 programs, illustrating basic programming techniques. Included are *Music Compiler Synthesizer, Advanced Ecosystem,* and *Steamboat Pilot*.

From the Reston Computer Group, 11480 Sunset Hills Road, Reston, VA 22090 (Phone 703-437-8900):

*Master Memory Map for the Commodore 64* guides the user through the inner workings of the 64, going into detail on creating sprites, producing sounds, and the like. Price is $14.95 (paper).

Not to lose the common touch, we'll close this Commodore books rundown with a bone toss to a small independent. *What's? for the 64* contains over 125 pages detailing software, hardware, books, and magazines for the 64, a magazine article bibliography, and a directory of sources supporting the 64 in all areas. Price is $15.00 plus $2.00 shipping. Write to 3494 Chickasaw Circle, Lake Worth, FL 33463.



© RAOUL TENAZAS 1984

# THE VIC CALCULATOR

## MULTIPLY YOUR COMPUTER'S MATHEMATICAL ABILITIES!

*By Joe Woyton*

This handy little program turns your VIC into a multifunction printing calculator with memory. The routine handles mixed chain calculations often found when preparing tax returns or balancing your checkbook or home budget.

The VIC operating system has excellent computing power *when under program control*. Unfortunately, the unprogrammed DIRECT mode of VIC operation is virtually useless except to evaluate individual expressions (like PRINT 3 + 5 RETURN). This program uses VIC BASIC string manipulation and VAL commands to input numbers (up to 8 characters), define operators (+,−,*,/,=), control number sign (+,−) and make screen notations like clear entry (CE), clear calculations (CC), etc.

These keys operate the VIC CALCULATOR:

| | |
|---|---|
| + | Addition operator |
| − | Subtraction operator |
| * | Multiplication operator |
| / | Division operator |
| = | Total |
| SPACE | +/−, change number sign |
| f1 | M+, add total to memory |
| f3 | M−, subtract total from memory |
| f5 | MR, memory recall |
| f6 | MC, memory clear |
| f7 | CE, clear entry, correct errors |
| f8 | CC, clear calculator |

When typing the program, pay close attention to the PRINT statements with TAB and CHR$ commands. These are used to format the screen display. They replace the often confusing CURSOR commands. If you have a printer, add the appropriate PRINT# statements at line 50 and 460 to generate hard copy.

If you don't want to do the typing, I will copy VIC tapes (only). Send $3.00, a blank tape, and a self-addressed stamped mailer. If you prefer, send $5.00 and I will supply the tape and return postage. Write to:

Joe Woyton
106 Braddock Drive
Mauldin, SC 29662

***SEE PROGRAM LISTING ON PAGE 78***

# SPREAD FOR THE C-64

*By Terry Silveria*

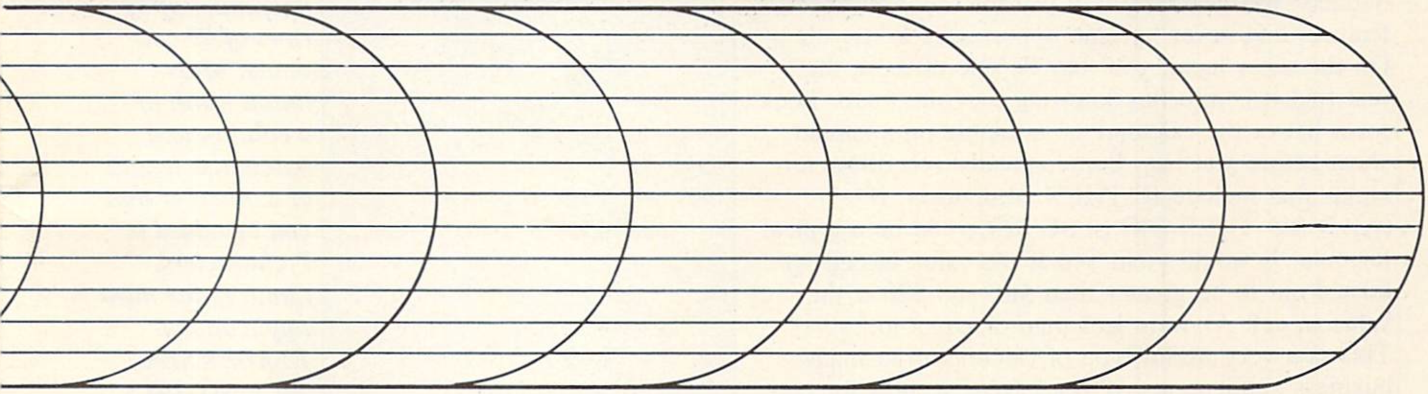
In the past it has been a rather simple matter to select a spreadsheet for the Commodore 64. With only a couple to choose from, it was hard to go wrong. Now there are over a dozen spreadsheets to compare, with capabilities that put the Commodore 64 in the same league as Apple and IBM. Along with increased spreadsheet selection and sophistication comes the difficulty of chosing the right one for your needs. A little background on spreadsheets before you start looking may help prevent that sick feeling of finding the perfect program after you purchase the wrong one.

Essentially, all spreadsheets set out with the same intention: to let you create charts of names and numbers and perform mathematical operations on those numbers. Most begin by displaying a similar screen. A row of numbers along the top corresponds to columns on the screen. Along the left edge alphabetic characters correspond to each row on the screen. The intersection of a column and a row is referred to as a cell and can be highlighted with a moveable cursor. Although the Commodore 64 displays a 40 character by 25 character screen, you can develop a spreadsheet that is much larger than that. To view other parts of the spreadsheet you move the cursor to the extreme edge and the entire sheet appears to move beneath the screen. It is a little like looking through a window. You never see more than a 40 by 25 character section at one time, but you can move the screen to view other sections of the sheet below it.

Using the keyboard the cursor can be moved in any direction from cell to cell. Once the cursor is positioned into a cell, data can be keyed into it. You can enter into a cell a name or a number or a formula. You could create a column of names corresponding to the expenses you have with your automobile. The column might include GAS, INSURANCE, REPAIRS and PARKING. You could then enter into an adjacent column the expected expense of each category: $45, $30, $10, $14. Not impressed yet? Well, the real power of a spreadsheet is in the formulas. Positioning the cursor in a cell below the list of expenses, directly below $45, you could enter a formula that totals the entire column. On one spreadsheet the entry would appear as SUM(B1-B4) where B1-B4 is referencing the cells comprising the column. What suddenly appears in the bottom cell is not the formula itself but the result. The formula will disappear from view when you move the cursor but can always be recalled if necessary.

And now for the real magic of spreadsheets. Move the cursor to one of the values, say, the gas expense of $35, and change it. The total value at the bottom automatically changes too. Change any of the values and each time the total is recalculated and displayed automatically. Applications of spreadsheets reach far beyond budgets and accounting. You can use the same program to create a sheet that will average your grades in school, monitor the fluctuations in the stock market, or track the sales of your staff. Once a spreadsheet has been developed, repetitive calculations in engineering or chemistry are as easy as keying in the variables. What impact on your savings over the next month will commuting to work on the bus have? Reduce the parking expense to $3, the gas expense to $10, and instantly a new total appears. You could have used a calculator, crossing out old values and inserting new ones and then recalculating each time. But with long lists of numbers and formulas that process becomes tedious and is prone to error. The convenience of changing original values to obtain

# SHEETS



new totals makes trial and error problems a breeze. Trial and error problems are very popular among financial planners and it is one of the reasons spreadsheets are so popular in the business community.

Armed with a little background on spreadsheets, what features should you look at in comparing them?

### *CONFIGURABILITY*

If you have read much about software or hardware, the word configurability has always been a menace. It is one of those buzz words that generally comes as a sort of warning that your peripheral or program may not work the way you what it to. Configurability is actually a reference to the flexibility and limitations of a device or program. In selecting a spreadsheet, you want to be aware of its configurability. You are not going to know exactly how many rows or columns you will need on applications you haven't even thought of yet, but compare the maximum for each spreadsheet. The maximum number of cells is another important statistic. If the software advertises a maximum of 256 rows, 100 columns, and 1000 cells, that does not mean you can have both 256 rows and 100 columns. That would work out to 25,600 cells. It means you can have up to 256 rows or 100 columns as long as you do not exceed 1000 cells.

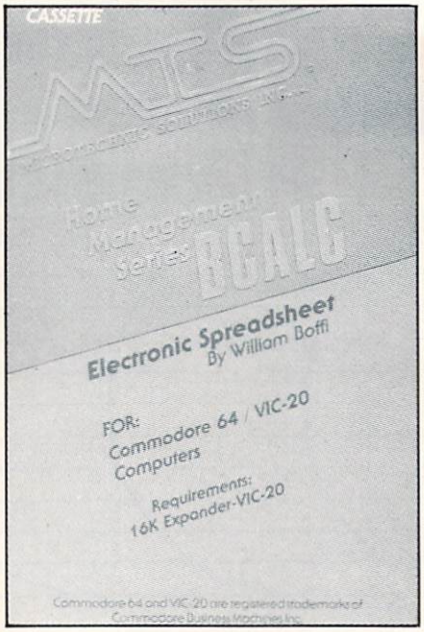
Look for a spreadsheet that states it is configured for your type of printer. Some are configured only for the 1525 and some other types of printers will give unpredictable results when printing. If it states it works with any properly configured printer you are probably safe. You are not going to get a stronger statement out of a software vendor. Some spreadsheets produce graphs that can be output to the printer. If you are looking for graphs be especially careful that the spreadsheet can be configured for your printer. The width of a column displayed on the screen varies with the different spreadsheets available. Most spreadsheets allow you to change the width up to some maximum value for the entire sheet.

### *EDITING CAPABILITIES*

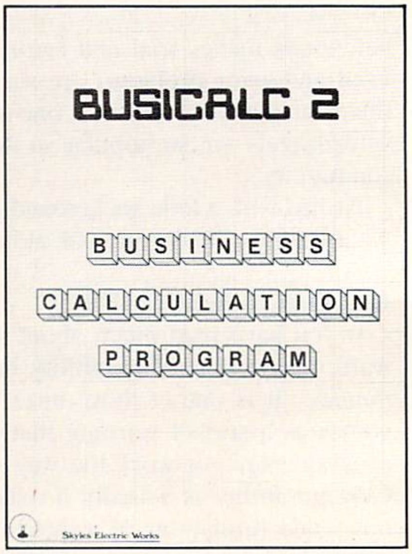
All spreadsheets allow you to enter values into a cell and to make corrections if necessary. But there are some handy little features that you will invariably wish you had that are available on some of the better programs. Row and Column copy commands are a must. If the column under January has 100 entries and is identical to the one that you must enter for February, about 80 entries down you're going to wish you had a column copy command: one that would copy all the entries from one column to another identically. If you have formulas in the column that make reference to the first column, you may want to have a relative change option in the copy command. This feature allows you to copy a column or row and automatically change any formula entry that references the column. If you are copying column 4 to column 6 and a formula references column 4, you may want it changed to 6 to reflect its new location. Some spreadsheets allow you to insert a column or row between two adjacent ones. This is a little tricky. If you insert a column between C and D what is the spreadsheet going to call it? Don't worry; they each work a little differently, but it can be done.

If you are entering lists of numbers under a row of names, as you move down the list the names will eventually scroll out of view. One useful feature is the ability to fix titles. This feature will allow you to hold a row or column in view regardless of

where you are viewing the spreadsheet.




***Microtechnic's* BCalc *is completely menu-driven. It allows up to 3500 cells with a maximum of 35 rows of 99 columns. Maximum width of a column and maximum length of a number that can be added is 7 characters. Limiting for most applications. READER SERVICE NO. 198***




**BusiCalc 2 *is the advanced version of* BusiCalc, *referred to below. It features a larger capacity, faster operating speed, and a save/replace feature.* BusiCalc 3 *is on the way; like* Jaws 3, *it will be 3D! READER SERVICE NO. 199***

## *CALCULATIONS*

All spreadsheets will allow you to add, subtract, multiply, and divide. Most go far beyond that, performing mathematical operations I can't even pronounce. You generally will pay for those advanced features and never have an opportunity to use them. On the other hand, you may be one of those that just find it comforting knowing they are there. Look for a list of the calculations available on a spreadsheet before you buy. Some spreadsheets allow formulas that include IF THEN commands. If (A1) > $10 THEN $40 ELSE $20 could be a typical formula. It would yield $40 if the value in cell A1 turned out to be greater than $10, and $20 if the value in cell A1 were less than or equal to $40. This is a very useful type of calculation in many business applications. If you have discounts for certain quantities sold, or a bonus on certain sales, this feature would facilitate the entries into a spreadsheet.

## *ADVANCED FEATURES*

Remember that generally, the more sophisticated spreadsheets require more effort in learning and application. Try to find a balance between versatility and ease of use. There are some advanced features that you may or may not desire. Remembering that the screen only displays 40 by 25 characters of the spreadsheet at a time, some spreadsheets allow you to split the screen to view two separate sections together. If you split the screen vertically, you can view two sections each 20 characters wide and move each section independent of the other.

One very powerful feature found in better software is a linking capability. Some spreadsheets allow you to create formulas that reference cells in other spreadsheets. You could create a home budget spreadsheet that would automatically look up values on your auto expense spreadsheet. Some spreadsheets allow you to display and print graphical representations of lists of numbers you create. Remember, different vendors have their own ideas about what a graph is. It may be anything from a row of asterisks to a high resolution multicolor display. Any elaborate printout of graphs requires that your spreadsheet be designed to work specifically with your printer.

Now that you're armed with some idea of what to look for in spreadsheets, the following is a brief overview of packages for the Commodore 64.

## *BCALC (FROM MICROTECHNIC SOLUTIONS)*

This spreadsheet allows up to 3500 cells with a maximum of 35 rows or 99 columns. This program is completely menu driven. Even formulas are menu driven. If you wish to create a formula you are prompted for row, column, or cell to be operated on. You are then prompted with add, subtract, multiply, divide, percent, percent change, total, average, high or low. After choosing the operation the calculation is made and the result placed in the cell of your choice. The maximum width of a column and the maximum length of a number that can be entered is 7 characters, which is very limiting for most applications. The program is in BASIC and is very sluggish to respond. Entering characters, as well as performing calculations, is slow. The printouts are configured for the 1525 printer or 1515 printers from Commodore, and any other printers can give unpredictable results. Microtechnic has sacrificed flexibility for ease of use.

## *BUSICALC (FROM SKYLES ELECTRIC WORKS)*

Spreadsheets can be developed with up to 99 rows

or 99 columns, as long as you do not exceed 1100 cells. The width of columns can be altered from 5 to 18 characters wide. Mathematical operations are limited to addition, subtraction, division, and multiplication. The program is in BASIC and is very sluggish. It seems to work with most properly configured printers, and you can change the format of the printout from that which you view on the screen. (A more advanced version, BusiCalc is now available, with additional capability; and the 3D BusiCalc 3 is on its way!)

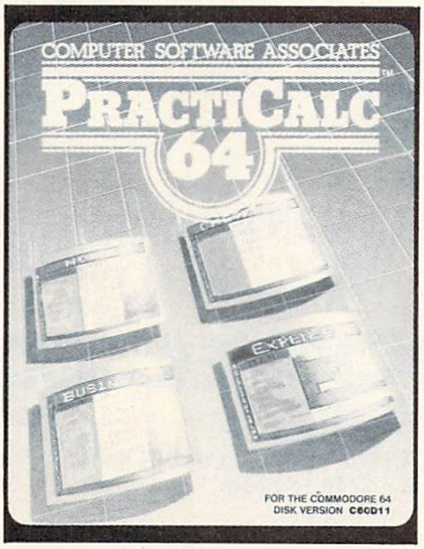
### *CALC RESULT EASY (FROM COMPUTER MARKETING SERVICES)*

This spreadsheet comes on a cartridge that inserts into the expansion port. Up to 254 rows or 63 columns not exceeding 1000 cells can be created. Columns can be configured from 5 to 18 characters in width. Formulas can be entered using addition, subtraction, division, multiplication, exponents, absolute values, square roots, logarithms, and natural logs. Formulas can include inequalities such as less than or greater than, and logical operators AND, OR, and NOT. Operations exist that will automatically calculate standard deviations, averages, random numbers, and the maximum or minimum values of a list. The IF THEN commands can be used within a formula. Bar charts can be displayed and printed to the 1525 printer. The program responds quickly to data entry and recalculations.

### *CALC RESULT ADVANCED (FROM COMPUTER MARKETING SERVICES)*

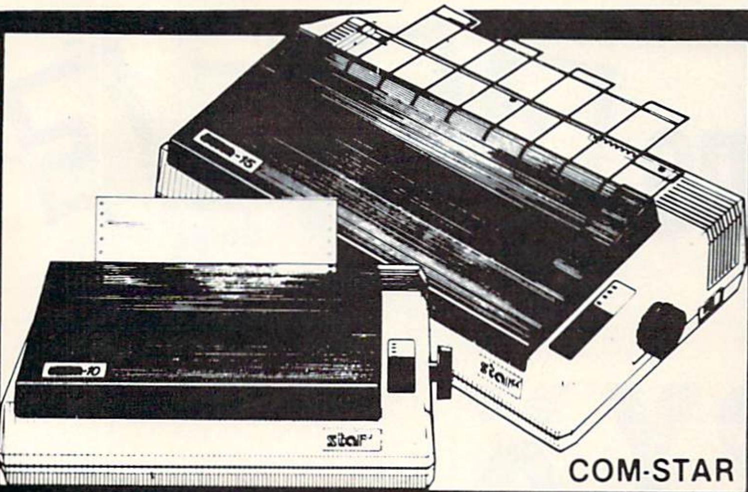
This version comes with both a cartridge and a diskette. It does everything *Calc Result Easy* does, and in addition allows you to split the screen to view separate parts of the same spreadsheet simultaneously. With *Calc Result Advanced* you can also merge different spreadsheets into one. Take the Jan-

*Continued on page 34*




**Practicalc** *lets you create up to 250 rows or 100 columns, as long as you don't exceed 2000 cells. Columns can be displayed from 3 to 30 characters in width. Lists can be alphabetically or numerically sorted.* ***READER SERVICE NO. 200***

# 80 COLUMN PRINTER SALE—$149.00*




## COM-STAR T/F

Tractor
Friction
Printer

only $199**

*•15 Day Free Trial -180 Day Immediate Replacement Warranty*

- Lowest Priced, Best Quality, Tractor-Friction Printers in the U.S.A.
- Fast 80-120-160 Characters Per Second • 40, 46, 66, 80, 96, 132 Characters Per Line Spacing
- Word Processing • Print Labels, Letters, Graphs and Tables • List Your Programs
- Print Out Data from Modem Services • "The Most Important Accessory for Your Computer"

### *STX-80 COLUMN PRINTER—$149.00

Prints full 80 columns. Super silent operation, 60 CPS, prints Hi-resolution graphics and block graphics, expanded character set, exceptionally clear characters, fantastic print quality, uses inexpensive thermal paper! Best thermal printer in the U.S.A.! (Centronics Parallel Interface).

### **DELUXE COMSTAR T/F 80 CPS PRINTER—$199.00

The COMSTAR T/F (Tractor Friction) PRINTER is exceptionally versatile. It prints 8½" x 11" standard size single sheet stationary or continuous feed computer paper. Bi-directional, impact dot matrix, 80 CPS, 224 characters. (Centronics Parallel Interface).

### Premium Quality—120 CPS COMSTAR T/F SUPER-10X PRINTER—$289.00

COMSTAR T/F (Tractor Friction) SUPER-10X PRINTER gives you all the features of the COMSTAR T/F PRINTER plus a 10" carriage, 120 CPS, 9 x 9 dot matrix with double strike capability for 18 x 18 dot matrix (near letter quality), high resolution bit image (120 x 144 dot matrix), underlining, back spacing, left and right margin settings, true lower decenders with super and subscripts, prints standard, italic, block graphics and special characters, plus 2K of user definable characters! The COMSTAR T/F SUPER-10X PRINTER was Rated No. 1 by "Popular Science Magazine." It gives you print quality and features found on printers costing twice as much!! (Centronics Parallel Interface) (Better than Epson FX 80).

### Premium Quality—120 CPS COMSTAR T/F SUPER-15½" PRINTER—$379.00

COMSTAR T/F SUPER 15½" PRINTER has all the features of the COMSTAR T/F SUPER-10X PRINTER plus a 15½" carriage and more powerful electronics components to handle large ledger business forms! (Better than Epson FX 100).

### Superior Quality SUPER HIGH SPEED—160 CPS COMSTAR T/F 10" PRINTER—$489.00

SUPER HIGH SPEED COMSTAR T/F (Tractor Friction) PRINTER has all the features of the COMSTAR SUPER-10X PRINTER plus SUPER HIGH SPEED PRINTING—160 CPS, 100% duty cycle, 8K buffer, diverse character fonts, special symbols and true decenders, vertical and horizontal tabs. RED HOT BUSINESS PRINTER at an unbelievable low price!! (Serial or Centronics Parallel Interface)

### Superior Quality SUPER HIGH SPEED—160 CPS COMSTAR T/F 15½" PRINTER—$579.00

SUPER HIGH SPEED COMSTAR T/F 15½" PRINTER has all the features of the SUPER HIGH SPEED COMSTAR T/F 10" PRINTER plus a 15½" carriage and more powerful electronics to handle larger ledger business forms! Exclusive bottom paper feed!!

### PARALLEL INTERFACES

For VIC-20 and COM-64—$49.00
For All Apple Computers—$79.00

NOTE: Other printer interfaces are available at computer stores!

### Double Immediate Replacement Warranty

We have doubled the normal 90 day warranty to 180 days. Therefore if your printer fails within "180 days" from the date of purchase you simply send your printer to us via United Parcel Service, prepaid. We will IMMEDIATELY send you a replacement printer at no charge, prepaid. This warranty, once again, proves that WE LOVE OUR CUSTOMERS!

Add $17.50 for shipping, handling and insurance. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. VISA — MASTER CARD ACCEPTED. We ship C.O.D.

**PROTECTO ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

SUPER-10" ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

*Buy 1 (at Sale Price!) Get One FREE

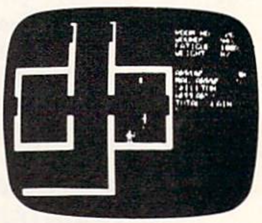
**Commodore 64**

# EPYX

# *Award Winning Games*

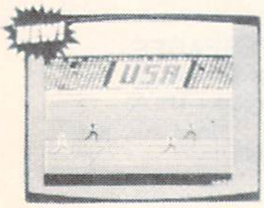


**Jumpman** — If you like Donkey Kong, you'll love Jumpman. Over 30 different screens with 8 speeds and 5 skill levels make this the fastest action game in the country. You must leap girders, climb ropes, and scale ladders to reach and diffuse bombs while avoiding robots, birds, bullets, explosives, crumbling walls, vanishing escape routes, and many other obstacles. (Truly a fantastic game!) List $39.95. **Sale $24.95** (Disk only).



**Temple of Apshai** — ***(computer game of the year)***. This is the standard by which other adventure games are judged. Full color graphics portray the temple and all its contents — magic, monsters, doomed cities and damsels in distress. Do battle in real time with over 20 monsters, expansion modules will keep your adventure alive in the future. List $39.95. **Sale $24.95**. (tape/disk).

*(Also Available For VIC-20)*




**Summer Games** — Now you can compete in the Olympics right along with the real competitors. Compete in track, swimming, diving, cycling, archery, weight lifting, rowing and more. This Olympic game is so realistic it even includes a beautiful opening ceremony and an awards presentation after each event. (Highly recommended, by far the best Olympics game we've seen). List $39.95. **Sale $28.00**. (disk only).



**Dragon Riders of Pern** — Based on the best selling book series by Anne McCaffey (over 10 million sold). This program forces you to make an alliance with the Lord holders, then mount your dragon (stunning graphics) and battle the dreaded thread. (Super fast action with a great theme). List $39.95. **Sale $24.00**. (tape/disk).




**Fax** — Stop wasting quarters! Now you can play Exidy's Fax at home. Play against the computer or an opponent to get the answer first, over 3,700 questions dealing with sports, television, movies, histories, and many more subjects will keep you playing for hours (a must for trivia buffs). List $34.95. **Sale $24.95**. (disk only).

* Buy Any Epyx Game From Protecto And Send The Proof Of Purchase Seals To Epyx And Epyx Will Send You A Free Game.

Add $3.00 for postage. Add $6.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. We accept Visa and MasterCard. We ship C.O.D.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

# Factory Clearance

## VIC-20 Cartridge Games

# CREATIVE SOFTWARE

**No. 1 Seller In U.S.A.**

**Sale**

**First Come Basis**

**Your Choice**

# $7.95

**Was $34.95**




**Astroblitz** — (Like Defender.) Voted No. 1 computer game overall by Creative Computing! Fantastic copy of an arcade classic. Destroy alien saucers and gun towers while avoiding enemy objects that float by. List $34.95. **Sale $7.95.**




**Serpentine** — Awarding winning game. Avoid hostile red snakes and lay your young while getting food to keep you alive. Fantastic strategy game that has become a classic. List $34.95. **Sale $9.95.**

**Priced Below Cost**

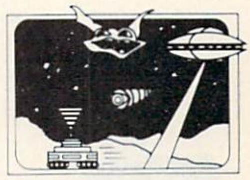



**Apple Panic** — The fast action computer game hit now comes to the VIC-20. You must dig traps in the brick floor and wait for the wandering apples to fall in. Then simply beat them on the head to push it through and destroy the apple. Otherwise the apple's will roll right over you and kill you. (Fast action.) List $34.95. **Sale $7.95.**

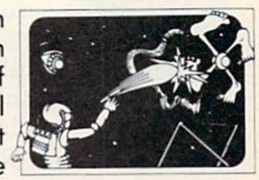
**While They Last**




**Rat Hotel** — Here you get to play the rat. Running and jumping from floor to floor in Hotel Paridisimo you must eat whatever you find but Waldo the maintenance man has set many traps for you that you must avoid. Plus Waldo has an atomic elevator that let's him jump several floors at once. See if you can get to the bottom before Waldo does you in. Fantastic graphics and sound. List $34.95. **Sale $9.95.**



**Terraguard** — Your mission is to destroy the aliens before their ship has a chance to land. Move your base along the ground and shoot up at them. List $34.95. **Sale $9.95.**



**Video Mania** — You're in an open field and strange alien creatures are roaming all over. If you stand still too long, they'll collide with you, causing instant annihilation. Your only defense is to destroy them by throwing your electronic, football-shaped alien zapper at them. **Sale $9.95.**




**Spills & Fills** — How much is too much? How much isn't enough? With this original concept educational game designed for grade schoolers through high school students, you can develop ratio and perspective skills. With three exciting and challenging skill levels, this program presents new and ever-changing challenges for the inquisitive mind. **Sale $9.95.**

**Trashman** (Better than Pacman). Dash for the trash and pick up CASH!!! List $29.95. **Sale $9.95.**




**Household Finances** — This 4 part program will take care of all your family's finances. Record, change and delete expense and income items from 16 categories, monthly and yearly totals are easily graphed and even a budgeting program to help you clamp down on excessive spending. List $34.95. **Sale $9.95.**

# PROTECTO
## ENTERPRIZES (WE LOVE OUR CUSTOMERS)

**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

Add $3.00 for postage. Add $6.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. We accept Visa and Master Card. We ship C.O.D.

Reader Service No. 216

# SANYO MONITOR SALE!!




## 9" Data Monitor

- 80 Columns × 24 lines
- Green text display
- Easy to read - no eye strain
- Up front brightness control
- High resolution graphics
- Quick start - no preheating
- Regulated power supply
- Attractive metal cabinet
- UL and FCC approved

- *15 Day Free Trial - 90 Day Immediate Replacement Warranty*

| | |
|---|---|
| 9" Screen - Green Text Display | *$ 69.00 |
| 12" Screen - Green Text Display (anti-reflective screen) | *$ 99.00 |
| 12" Screen - Amber Text Display (anti-reflective screen) | *$ 99.00 |
| 14" Screen - Color Monitor (national brand) | *$249.00 |

*PLUS $9.95 for Connecting Cable.

### *Display Monitors From Sanyo*

With the need for computing power growing every day, Sanyo has stepped in to meet the demand with a whole new line of low cost, high quality data monitors. Designed for commercial and personal computer use. All models come with an array of features, including upfront brightness and contrast controls. The capacity 5 × 7 dot characters as the input is 24 lines of characters with up to 80 characters per line.

Equally important, all are built with Sanyo's commitment to technological excellence. In the world of Audio/Video, Sanyo is synonymous with reliability and performance. And Sanyo quality is reflected in our reputation. Unlike some suppliers, Sanyo designs, manufactures and tests virtually all the parts that go into our products, from cameras to stereos. That's an assurance not everybody can give you!

SANYO

*Official Video Products of the Los Angeles 1984 Olympics*

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 218

**THE DIRECTORY OF SOFTWARE PUBLISHERS: HOW AND WHERE TO SELL YOUR PROGRAMS, edited by Eric Balkan (Van Nostrand Reinhold Company, 1983; $25.50). 310 pages, hardbound.**

Have you ever thought about trying to sell a program you wrote? *A Directory of Software Publishers* offers guidance in marketing software as well as 1000 names, addresses, and phone numbers of companies that buy original software. These companies range in size from computer magazines that buy program listings to superstars such as Sirius and Broderbund. While most of the companies support small systems, some publish for computers as large as mainframe IBM's.

Balkan compiled a list of relevant questions and sent them to software publishers. About 165 responded, and their responses form the central directory in the book. The innocent who has not previously surveyed the shark-filled waters of the free market may find the questions alone worth the price of the book. The questionnaire profiled possible legal and economic considerations for the software author.

The questions begin with essential information about the publisher. How long has the company been around? Is software publishing its only business? How many packages does it have on the market? How many submissions does it receive a week, and what percent does it accept?

What about the programmer? If the company likes his program, do they buy outright or offer royalties? Will they give the author help in programming and documentation? Who owns the copyright when a deal is concluded? How long will it be before the programmer gets a response on a submission?

Every company does not answer all these questions or the others in Balkan's questionnaire. Some offer additional comments on formatting or documentation requirements. The person who has a program to sell can make an informed choice on where to send it.

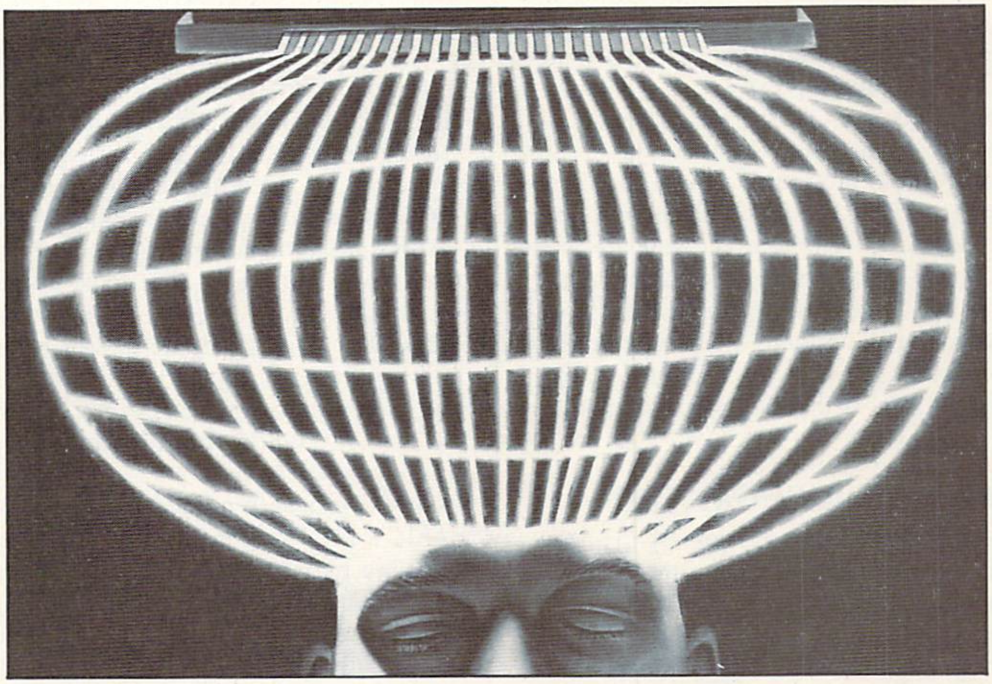
A second directory, of companies which either did not respond to the questionnaire or were not sent one, lists over 800 names, addresses, and phone numbers. Balkan also includes a brief description of the kind of software the company advertises. While this directory is not as useful as the other, it is a formidable compilation. Probably no other single source has so much information on software publishers. Publishing lag means that some of this information may be a year or more out of date, but most of it should be current.

For the novice at marketing, Balkan offers reprints of several useful articles. They are not detailed, but do offer guidelines for putting together a software package for presentation. Short discussions on legal protection of programs, selling spoftware directly to users, and Chris Crawford's advice on writing game programs round out the package.

Anyone interested in selling programs needs to be familiar with this book. It is a possible first step in making dreams come true.

*—Annette Hinshaw*

# MEMORY MANAGEMENT



© STEVE CIESLAWSKI 1984

# ON THE VIC 20 AND C-64

*By Morton A. Kevelson*

Now that you know BASIC, do you find yourself running short of memory on your VIC 20? Do you wish you could somehow use that extra three-kilobyte expander you bought way back when, but don't know machine language? How about you Commodore 64 users? Did that last database program you wrote leave you short on memory for the data you wanted to enter? Well, read on. For the VIC 20 we will show you how to get up to 39,423 bytes for BASIC. For the Commodore 64 we will have 43,007 bytes available for BASIC.

The VIC 20 in its base configuration, with only 3583 bytes available for the storage of BASIC programs, is somewhat limited. This fact becomes obvious to most users shortly after they obtain more than a minimal mastery of BASIC. It is no wonder that the most popular first accessory for VIC 20 owners is some form of memory expansion. These are available in several configurations, starting with three kilobytes minimum and increasing in eight kilobyte increments for a maximum expansion of 24 kilobytes. The last configuration results in 28,159 bytes free for BASIC. For many users, the first add-on module is the three kilobyte expansion, which is shortly followed by an eight kilobyte module. These same users are subsequently disappointed when they discover that VIC BASIC is not able to simultaneously utilize the three kilobyte and eight kilobyte expanders. Adding one of the available memory expander boards is no help in this situation.

The techniques presented here will allow VIC 20 BASIC programmers to regain the use of that disappearing three kilobytes. This will be accomplished by using some simple BASIC commands. A knowledge of machine language will not be required. As an added benefit, these same techniques will allow VIC 20 users to expand the capacity of their machines to a grand total of 39,423 bytes available for BASIC. That is a half kilobyte more than the 38,911 bytes that Commodore 64 users start out with.

Speaking of the Commodore 64, these same

techniques will allow the addition of four kilobytes to BASIC for a total of 43,007 bytes. Moreover, for the Commodore 64, this additional memory is absolutely free. There is nothing extra to buy, as the required RAM is already built into the machine.

To begin with, let us take a look at some memory maps. These are nothing more than simple charts showing the way the parts of memory are used by the computer. Different versions are presented in the manuals that come with your computer and are published in general articles on the workings of the machine. The maps we present here will be slightly different, as we will be emphasizing some unique features of the Commodore operating system.

## COMMODORE 64 MEMORY MAP

We will start with the Commodore 64 memory map. As it turns out, for our purposes it is simpler than that for the VIC 20. The memory map for the latter is complicated by the movement of the screen memory configurations. For the purposes of this discussion, the Commodore 64 should not have any ROM cartridges plugged into the expansion port, as they will affect the initial memory configuration.

The Commodore 64 memory map we will be using is shown in Figure 1. It is nothing more than a diagrammatical representation of the memory allocation in the computer. Memory address 0 has been placed at the bottom of the drawing following the convention of the *Commodore 64 Programmer's Reference Guide*. The first kilobyte (or 1024 bytes) of available RAM is used by the operating system and the BASIC interpreter. There are several memory locations in this area which are crucial for our application. These will be discussed in considerable detail later on. The next kilobyte is devoted primarily to the screen display and will not affect our intended application. Starting at address 2048 and continuing up to address 40959 is where BASIC programs are normally stored. Mathematically oriented readers will note that this is a total of 38,912 bytes of RAM, one more than the 38,911 bytes free as shown by the power up message. The missing byte is location 2048, which must always contain a zero marking the start of a BASIC program. As such, Commodore leaves this byte out of the initial memory count.

This is probably a good place to talk about BASIC's FRE command which tells you how many bytes are remaining (free) for BASIC storage. Since this command returns a value which has to be put somewhere, trying to execute it in immediate mode often results in a SYNTAX ERROR. The proper way is to execute

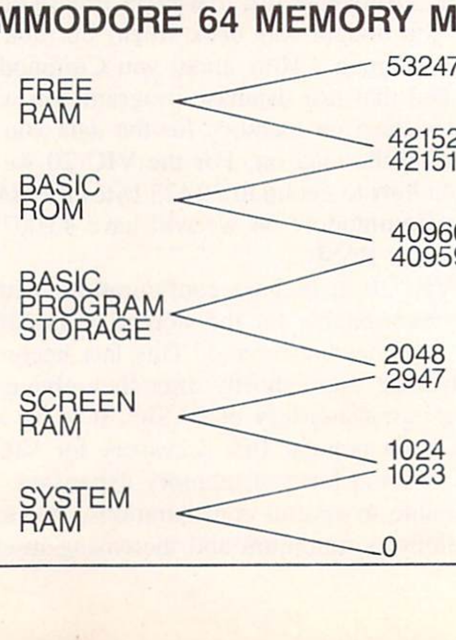
```
PRINT FRE(0)
```

in immediate mode. The use of the PRINT command allows the returned value to be placed on the screen. The zero in the parentheses is a dummy variable and could be any integer. Commodore 64 users when executing this command will get a response of −26627. This is because the FRE command calculates the amount of available memory as

### TABLE 1
### BASIC MEMORY MANAGEMENT ADDRESS

| MEMORY ADDRESS | FUNCTION | CONTENTS VIC 20 | CONTENTS C-64 |
|---|---|---|---|
| 43<br>44 | Start of<br>BASIC | 1<br>18 | 1<br>8 |
| 45<br>46 | Start of<br>Variables | 3<br>18 | 3<br>8 |
| 47<br>48 | Start of<br>Arrays | 3<br>18 | 3<br>8 |
| 49<br>50 | End of<br>Arrays + 1 | 3<br>18 | 3<br>8 |
| 51<br>52 | Bottom of<br>Strings | 0<br>64 | 0<br>160 |
| 55<br>56 | End of<br>Memory | 0<br>64 | 0<br>160 |

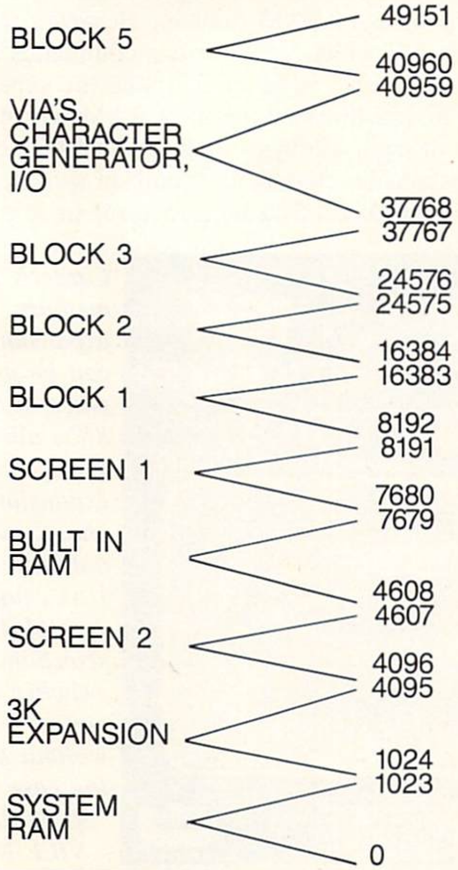
### FIGURE 1
### COMMODORE 64 MEMORY MAP



an unsigned, two byte integer value. This means that the result has to be subtracted from 65,536 to get the correct answer. Doing so results in 38,909 bytes free. To this you might ask: where did the other two bytes go, as we have not even gotten started? The missing bytes have been reserved for the first BASIC line number. We will see more about this later.

Getting back to Figure 1, the next eight kilobytes, from addresses 40960 to 49151, are occupied by the BASIC ROMs. These are the actual machine language programs which allow the Commodore 64 to understand the BASIC programs we type in. In the case of the Commodore 64, there is actually additional RAM hidden "underneath" this ROM. We cannot use this RAM since BASIC cannot fully access it without some machine language. Above the BASIC ROMs, from addresses 49152 to 53247, are four kilobytes (4,096 bytes) of free RAM. It is this free RAM which we will access for BASIC programs. Note that this RAM is normally used for various machine language routines such as the Commodore 64 DOS wedge. Any of these routines cannot be used while this RAM is being used for BASIC.

**FIGURE 2**
**VIC 20 MEMORY MAP**




## *VIC 20 MEMORY MAP*

The VIC 20 has several possible "normal" memory maps because of the different amounts of memory which can be added at any one time. The configuration we are discussing will be using the three kilobyte expander cartridge and at least one eight kilobyte expander cartridge. Note that the three kilobyte expander cartridge is essential for this application. In addition, some form of expansion chassis is required. This piece of hardware, when inserted into the VIC 20 expansion port, adds two or more expansion sockets to the computer.

The particular configuration of the VIC 20 memory map we will be using is shown in Figure 2. This particular configuration results when at least an eight kilobyte expander cartridge has been added to the expansion port. The presence of the three kilobyte cartridge does not change any of the numbers. Starting at the bottom with address 0, the first 1024 bytes are used by the BASIC interpreter and the operating system. As with the Commodore 64, several of the locations in this range are crucial for our application. The next three kilobytes, from addresses 1024 to 4095, are normally vacant on the unexpanded VIC 20. The three kilobyte memory expander, which is required for this application, will fill this group of addresses with RAM. The next four kilobytes from addresses 4096 to 8191 contain the built-in RAM that the VIC 20 is supplied with. In this area, 512 bytes are devoted to the VIC 20 screen display. When the eight kilobyte RAM module is in place, the screen will be located from addresses 4096 to 4607 as shown (Screen 2). If there is no memory expansion or just the three kilobyte expander in use, the screen will move to the top of this block at addresses 7680 to 8191 (Screen 1).

It is this "floating" screen which prevents BASIC from using the three kilobyte expansion when an eight kilobyte module is in use. This is because BASIC normally requires a continuous or uninterrupted block of RAM for program storage. Hardware limitations on the VIC 20 confine the screen RAM to somewhere in the built-in four kilobytes of RAM. As a result, the screen RAM just gets in the way of the three kilobyte module. We will shortly see how to get around this restriction.

We can now see where the 3583 BYTES FREE message comes from when the unexpanded VIC 20 is turned on. The built-in RAM from addresses 4096 to 8191 yields 4096 bytes of which 512 bytes are reserved for the screen display, leaving 3584 bytes for BASIC. The first byte for BASIC must always be a zero which leaves 3583 bytes free. If you execute in immediate mode,

```
PRINTFRE(0)
```

at this point, you will see 3581 bytes free, as two bytes have been reserved for the first BASIC line number.

Returning to Figure 2, starting at address 8192 and continuing up to address 32767 there is another gap in the memory. This gap can be filled with up to three eight kilobyte modules for a total of 24,576 bytes of expansion RAM. Adding this to the built-in RAM results in 28,159 bytes free maximum normally available. The first eight kilobytes of this range (addresses 8192 to 16,383) have been designated as Block 1 on our map. The next two eight kilobyte sections have been designated as Block 2 and Block 3 respectively. In this context, the first eight kilobytes (addresses 0 to 8191) can be referred to as Block 0.

Continuing upwards, we encounter an eight kilobyte section which contains the Character Generator ROM, the screen color map, the input/output chips, and some locations reserved for future input/output expansion such as an IEEE-488 interface. These locations are of no direct interest for our purposes. This area has been designated as Block 4 on the map. The next eight kilobytes are rather interesting. The memory locations from address 40960 to 49151, designated as Block 5, were intended by Commodore for the use of game cartridges. The operating system actually checks for the presence of a game cartridge at this location on power up. If one is found, the operating system turns control over to the cartridge.

The Block 5 memory locations will be of considerable interest for us. We will show how to place eight kilobytes of RAM into this location and link this memory to BASIC.

## *VIC 20 EXPANSION RAM*

Expansion memory can be added to the VIC 20 in several ways. Commodore makes RAM available in three cartridge formats: a three kilobyte, an eight kilobyte, and a sixteen kilobyte cartridge. To achieve maximum memory expansion using Commodore cartridges, you will need one three kilobyte cartridge, one sixteen kilobyte cartridge, and two eight kilobyte cartridges. Since the VIC 20 has a single expansion slot on the back, you will also need an expansion chassis with a minimum of four slots. These are available from a number of different manufacturers. The two eight kilobyte cartridges are required because Commodore has put block selection switches on these. These switches are accessed by opening up the cartridge as per the accompanying instruction sheet. The block selection switches have been left off Commodore's sixteen kilobyte cartridge. This cartridge will always occupy Block 1 and Block 2 of our memory map. The block selection switches on the eight kilobyte cartridges should be set for Block 3 and Block 5.

A single sixteen kilobyte cartridge can be substituted for the two eight kilobyte cartridges if it is equipped with block selection switches. This would allow the use of an expansion chassis with only three slots. The Cardco sixteen kilobyte cartridge is equipped with block selection switches which can be set without opening the case.

The Super Expander cartridge for the VIC 20 has three kilobytes of RAM built in. However, the cartridge ROM, which adds various commands to BASIC, is located in block 5. Using the super expander to add three kilobytes of RAM will prevent the use of eight kilobytes of RAM in Block 5. Some expansion chassis have built-in switches which allow Block 5 to be turned off in selected




*Cardco's 16K memory expansion cart can be used in place of two 8K's, allowing use of an expansion chassis with only three slots. It is equipped with block selection switches that can be set without opening the case.* *READER SERVICE NO. 184*

slots. Using one of these will allow the Super Expander functions to be turned off while still leaving three kilobytes of RAM active. This procedure would allow the use of the eight kilobytes of RAM in Block 5.

The most cost-effective approach is to use one of the expansion chassis that have all the RAM built in. The chassis should be equipped with 35 kilobytes of RAM. These chassis are usually equipped with one or two additional slots for cartridges. There should also be a number of switches on the board to allow switching various blocks of RAM in and out. This allows you to avoid memory conflicts as described for the Super Expander.

## *BASIC MEMORY MANAGEMENT*

The key to BASIC's memory allocation is in the contents of several memory addresses. The particular locations are the same for the VIC 20 and the Commodore 64. These locations and their contents, immediately after power up, are shown in Table 1.

At this time we will try an experiment. Type in and save the short test programs shown here, using the version suitable for your machine. Save both programs on the same disk or tape before running either. If you are using cassette, be sure to save TEST 1 on tape ahead of TEST 2. Also, leave out the ",8" after the LOAD commands. If either program is interrupted by a syntax error, the computer should be reset and the saved version reloaded before corrections are made. This is necessary as the memory locations of Table I will be changed by each program, which could cause some strange results. Turning the computer off and on can be avoided by executing

```
SYS64802
```

for the VIC 20 or

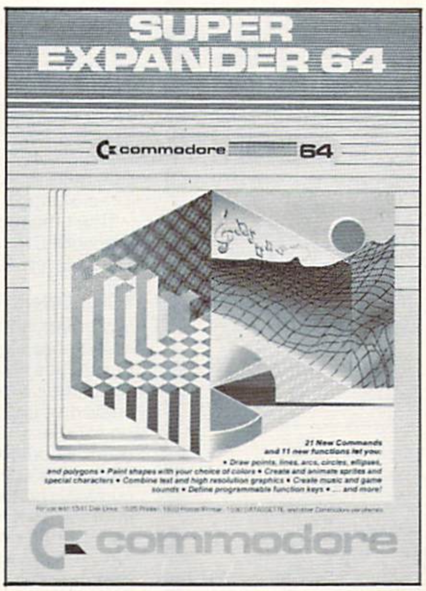
```
SYS64738
```

for the Commodore 64. Remember to save both programs before running as this procedure will make them inaccessible. Do not try to run TEST 2 by itself, as it must be loaded by TEST 1.

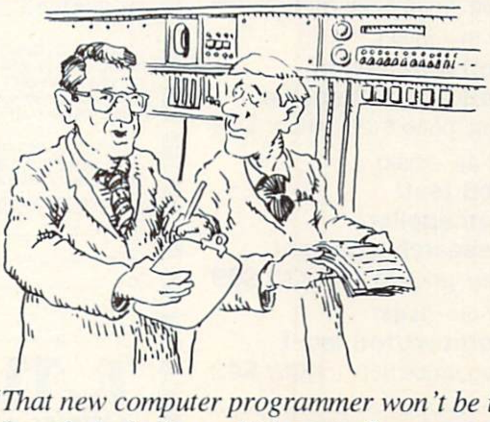
If you are using the VIC 20, you will need the three kilobyte RAM cartridge and at least eight kilobytes of RAM in Block 1. When TEST 1 is run it will print out a message and the contents of the memory locations of Table 1. It will then LOAD and execute TEST 2, which will do the same. The significance of this operation is that TEST 2 will be loaded into and executed from the three kilobyte RAM expansion for the VIC 20. For the Commodore 64, TEST 2 will be loaded into memory starting at location 40960. After TEST 2 prints out the message which has been passed to it from TEST 1, it will print out the new values in the memory locations of Table 1. Control will then be transferred back to TEST 1, proving that it is still in memory.

That's all we have room for this time. Next month we'll explain how the test programs work, the significance of the memory locations in Table 1, and just where the different parts of a BASIC program are stored in memory. We will also show how to load and run up to three different BASIC programs at the same time in the VIC 20 (two for the Commodore 64). We will also explain how data and variables are passed between these programs. □

*SEE PROGRAM LISTING ON PAGE 79*




*Super Expander 64 is Commodore's answer to the problem of 64 memory expansion. It provides 21 commands that facilitate graphics and sprite creation, use of music and game sound, etc. READER SERVICE NO. 185*



*"That new computer programmer won't be in today—he's feeling a little out of sequence."*

# SPREAD-SHEETS

*Continued from page 21*

uary budget and combine it with the February budget to get a single spreadsheet displaying both months. You can also hold two different spreadsheets in memory and flip back and forth between them. You can view January's budget on the screen, press a single key, and immediately view February's.

## *ESP>CALC (FROM NEW LEAF INC.)*

Up to 645 rows or 645 columns can be created on a single spreadsheet as long as you don't exceed 2000 cells. This program is designed around ease of use. You are prompted in the beginning for the number of rows and columns for your spreadsheet. You can perform addition, subtraction, division, multiplication, percent added, percent subtracted, sum, and average minimum or maximum. To insert new rows you must reformat the spreadsheet and copy a previous one. Printing is configured for the 1525 printer. Again, versatility is compromised for ease of use, with data entry and calculations slow.

## *MULTIPLAN (DISTRIBUTED BY HES)*

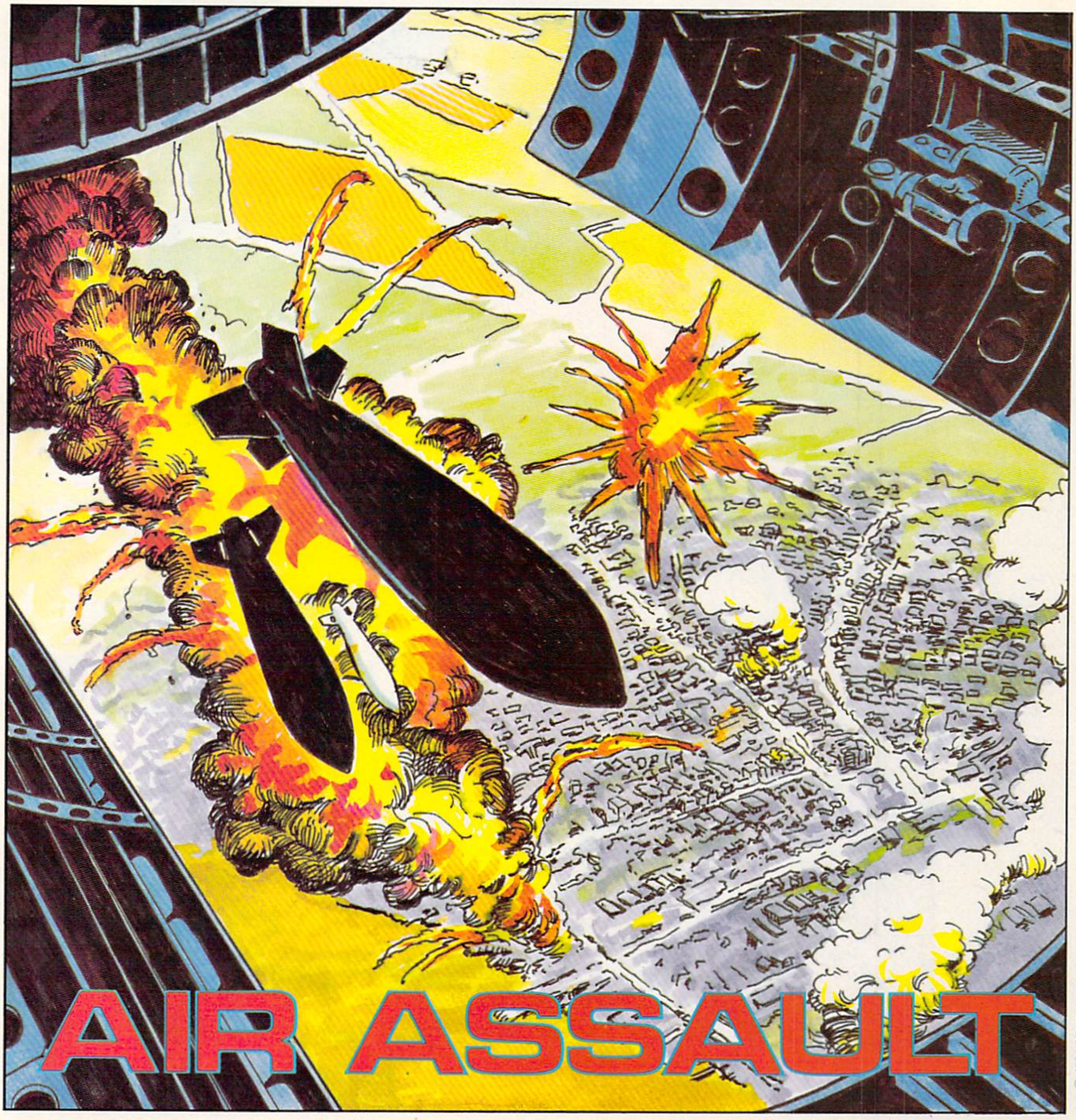
Create up to 255 rows or 63 columns as long as you don't exceed the 10K of memory within the workspace. This is a professional quality spreadsheet. You can give individual cells names and refer to them in formulas by name rather than row and column number. Formulas begin to appear in English rather than mathematics. The formula PRICE + TAX = TOTAL can be used rather than A1 + B1 = C1. *Multiplan* has a split screen option as well as linking capability. You can create formulas that reference information in other spreadsheets. There is not enough room in this article to mention all of the mathematical operations performed: the standard addition, subtraction, multiplication, and division, as well as trigonometric function, etc. Documentation is lengthy but well written.

## *OMNICALC (BY HES)*

Up to 255 rows or 65 columns within internal memory constraints. Columns can be displayed from 4 to 20 characters in width.

There are a number of predefined formulas. The user picks the formulas and is prompted for cells, rows, or columns to be operated on. A linking facility lets you to use values from other spreadsheets. In addition to a surplus of standard mathematical operations (addition, subtraction, division, exponential, sum, log, etc.) there are some interest-

*Continued on page 93*




Art © 1984 Vincent Waller

# AIR ASSAULT

*By Bob Lloret*

You are chief gunnery officer in charge of protecting your city against enemy air assaults. Perched high above your city, you know that at any minute a barrage of bombs will fall from the sky. The citizens are depending on you to intercept the bombs before they hit.

*Air Assault* is a program designed for one player with joystick. After typing in the program, save a copy to disk or tape before running in case of any typing errors. Type "RUN" and the title screen will be displayed. Press "fl" when you're ready for action.

After pressing "fl" the message "Hold on. . . I'm building the city" will appear on the screen with a short pause. During this time, the program is moving character sets and creating the information needed to run the program.

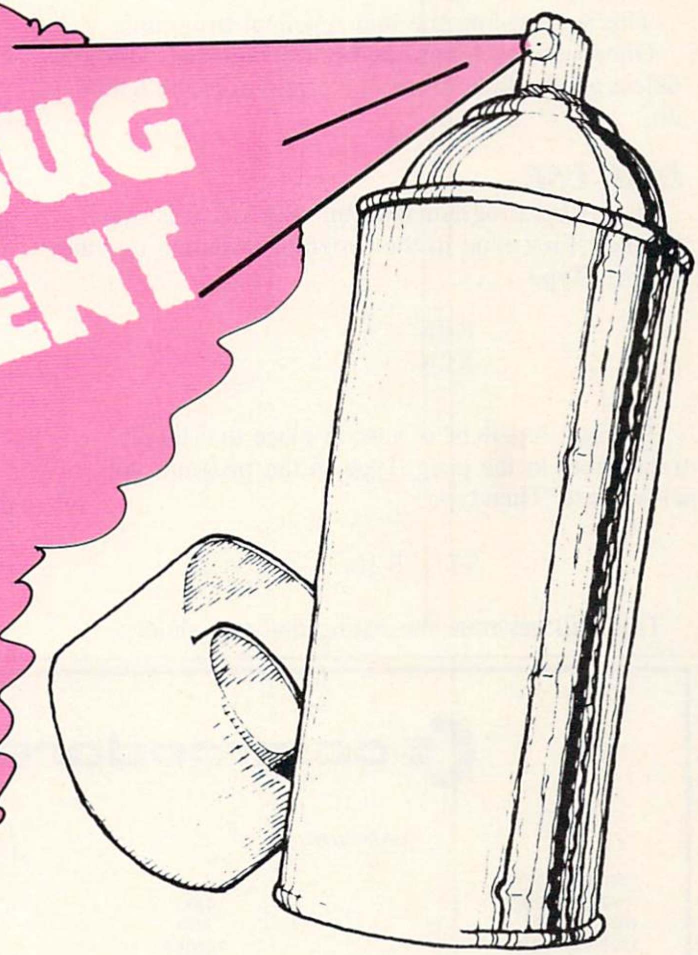
The next screen shows a city at the bottom of a narrowed screen, with your gun sight in the lower middle portion of the screen. On the right side you see the words "Bomb" and "Score." This will keep track of the number of bombs that fall on your city and the score you attain for intercepting the bombs. The higher on the screen you stop the bomb the

*Continued on page 71*

*By Michael Kleinert and David Barron*

# VIC 20 BUG REPELLENT

## An easy-to-use proofreader for Ahoy! VIC 20 programs



Last month we gave you 12 oz. of bug repellent for your Commodore 64. This month we give you the same, but for his brother VIC. The VIC version is basically a no-frills version of that for the 64. Some corners had to be cut in order to fit this program into the VIC.

The C-64 version situates itself at a place in memory where it won't bother anyone. Unfortunately, this couldn't be done with the VIC program: we placed it in the cassette buffer. If you use a disk drive, don't worry; everything is easy. If you use a cassette, things are not so simple. Let's start with the procedure for using the cassette version, for it is the more difficult of the two.

### *CASSETTE USE*

First you must type in the program in listing one. Save this program, for you will use it every time you wish to employ our method for checking your programs. Now type in the program you wish to check.

(Note: our program uses lines greater than 63000, so please make sure your program doesn't!)

When done, your program and ours will coexist in the machine with our program being the end of yours. Save your program, and you are ready to disinfect it. Type

```
RUN 63000
SYS 828
```

The computer will now generate the list of line numbers and corresponding codes for you to compare with the list in *Ahoy!* Remember, if there is a discrepancy an error exists in that line.

You may type SYS 828 as many times as you wish, but if you use the cassette for anything remember to type the following:

```
RUN 63000
```

This will restore the bug repellent program.

Once your program has been disinfected, you may delete all the lines from our program (Line 63000 on).

### *DISK USE*

Using this program with the disk drive is much simpler. First type in the program in listing one and save it. Type

```
RUN
NEW
```

The bug repellent is now in place and totally transparent to the user. Type in the program you wish to use. Then type

```
SYS 828
```

This will generate the listing that you should compare with the listing in *Ahoy!*

This is all there is to using the disk version. Cassette people, *buy disk drives!*

### *GENERAL NOTES*

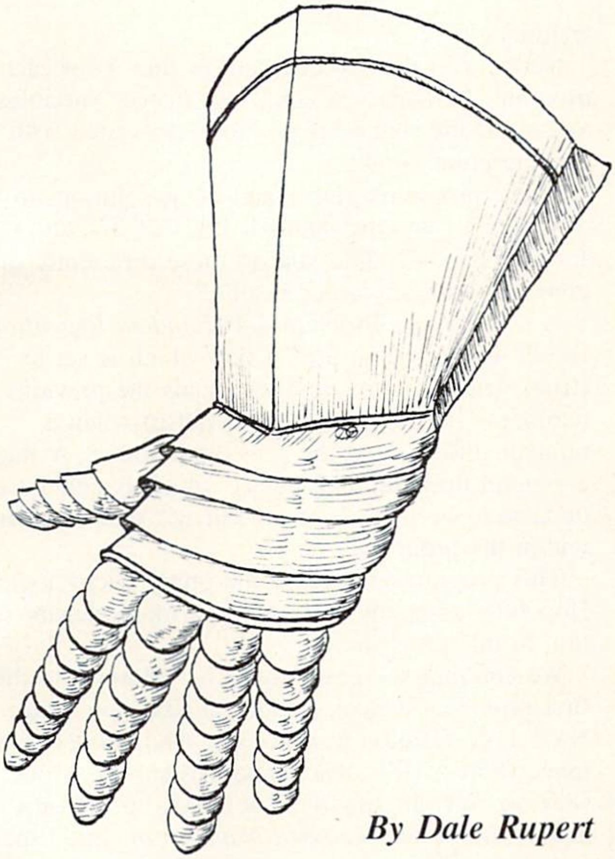
This version does not ask you whether you wish to send the list to printer or screen. This was eliminated because of memory problems. If you wish to send the list to the printer type

```
OPEN 4,4
CMD 4
SYS 828
CLOSE 4
```

A pause feature is built into the program. To pause the listing, depress the shift key. To permanently pause it, depress shift lock. To continue, release shift lock.

*SEE PROGRAM LISTING ON PAGE 77*

# COMMODARES



*By Dale Rupert*

Each month, we'll present several programs designed to toggle the bits in your cerebral random access memory. We invite you to send your solutions to *Commodares*, c/o *Ahoy!*, 45 West 34th Street—Suite 407, New York, NY 10001. We'll print the cleverest, simplest, shortest, or most unusual solutions. Your original programming problems would be equally welcome!

## *Problem #5-1: Faster Computer*

Which is the faster computer, you or your Commodore? See if you can determine the output of this program in less time than it takes to enter and run the program on your computer.

```
10 DIM M(999),N(999)
20 FOR J=1 TO 999
30 M(J)=1000 - J
40 N(J)=J
50 NEXT J
60 FOR J=1 TO 999
70 SUM=SUM + M(J) + N(J)
80 NEXT J
90 PRINT SUM
```

Once you see what is happening here, you should be able to solve the problem faster than the computer can.

## *Problem #5-2: Random Array*

Can you figure out what the values in line 100 represent?

```
10 DIM A(100)
20 FOR J=1 TO 100
30 A(J)=RND(1)*100
40 NEXT J
50 L=1 : S=1
60 FOR J=2 TO 100
70 IF A(J) > A(L) THEN L=J
80 IF A(J) < A(S) THEN S=J
90 NEXT J
100 PRINT A(S) , A(L)
```

## *Problem #5-3: Running Average*

Write a program which calculates and displays the running average of a set of numbers typed in by the user. For example, the left column shows the number entered, and the right column shows the current running average:

| | |
|---|---|
| 4 | 4 |
| 8 | 6 |
| 3 | 5 |
| 9 | 6 |

After the 4 is typed, the total is 4 and the average is 4/1 or 4. When the 8 is typed, the total is now 12 (4+8) and the running average is 12/2 or 6. Similarly, the 3 brings the total to 15 so the running average is 15/3 or 5. The 9 makes a total of 24 and a running average of 24/4 or 6.

If this is not much of a challenge for you, modify your program so that the user can delete any of the numbers typed, and the running average is updated accordingly.

## *Problem #5-4: Letter Triangle*

Write a program which displays a triangular-shaped block of letters with one "A" at the top, two "B"'s in the second row, etc.... down to

twenty-six "Z"'s in the bottom row. It starts like this:

```
A
BB
CCC
DDDD
```

Problem 4-3: *Wrong Number* from last month's *Commodares* is easily corrected by changing the formula in line 30 to ASC(L$) — ASC("A") + 1. The advantage of using this formula is that you don't have to know or look up the ASCII value of the letter "A". If you did know that ASC("A") equals 65, then the formula in line 30 could have been simply ASC(L$) — 64.

On page 91 are solutions to the other three problems from last month. In the listing for Problem 4-1: *Squared Sum*, notice how each number is separated into its digits by first converting the number to a string variable in line 20. Then the value of each digit is obtained in line 40 with the use of the MID$ function. The solution shown for Problem 4-2: *Sum of Cubes* uses the same technique.

Notice also the test criterion in line 70 of each program. Perhaps you could use integer variables and avoid the round-off problem associated with floating-point numbers.

These programs give 1 and 81 as solutions to Problem 4-1 and the values 1, 153, 370, 371, and 407 for Problem 4-2. Did you do these differently or come up with any other results.?

A solution for Problem 4-4: *Random Repetition* is listed. The program uses a flag which is set to −1 (true) whenever one number equals the previous number. The flag is reset to 0 (false) when a number differs from the previous number. A flag is a general programming device which is either true or false to keep track of the current state of affairs within the program.

This program involves some pretty tricky logic. Hopefully someone will submit a more elegant solution to this program.

We continue to receive readers' solutions to the first problems. Chuck McGaffin (Ballston Lake, NY), D.G. Gannon (Anchorage, AK), Robert Lawrence (Forest Hills, NY), and William G. Mork (Racine, WI) all submitted solutions to both the *Square Name* and *Crossed Name* problems. Linda Koiro (Northfield, NJ) solved the *Square Name* problem. The solutions ranged from straightforward to sophisticated.

A couple of interesting notes from their solutions: Robert Lawrence's solution was written for the TRS-80 Model 1. Programmers of all types of computers are welcome to try their skills on the *Commodores*! William Mork used POKEs to two memory locations to position the cursor. The following line can be used in your program to print "ABCDEF" starting at column 8 of row 15 on the screen:

```
POKE 214,14 : PRINT : POKE 211, 8
: PRINT "ABCDEF"
```

The first POKE and PRINT move the cursor to line 15 (one more than the value POKEd into address 214.) The POKE to address 211 moves the cursor over to column 8 on the current line where the "ABCDEF" is printed. Remember that the rows range from 0 to 24 and the columns go from 0 to 39. This looks like a handy way to position the cursor. Thanks for the tip, William! To all readers, good luck on this month's *Commodares!*

***SEE PROGRAM LISTINGS ON PAGE 91***




ARTIST: STEVEN CIESLAWSKI

# Educational Software:
## a guide for parents

Part III

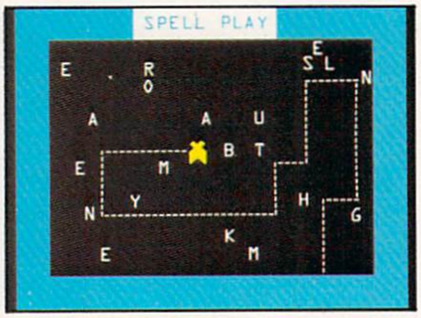
*By Richard Herring*

Few educational computer programs available in schools teach concepts. Most are drill and practice or multiple choice oriented. Further, program objectives usually ignore high order skills like comprehension, application, synthesis, and evaluation. Recalling previously learned facts is the major emphasis. These are some of the findings from a recent study by Vickie Blum, a researcher at Columbia University's Teachers College.

To help you avoid a similar situation in your home, this series of articles can serve as a guide. Each month, we will consider several factors on which to judge programs before you buy. Maybe we can even influence an aspiring software author or two. If something you read here rings true or strikes a discordant note, I'd really like to hear from you.

### *EASE OF USE*

Educational software should be simple to load.

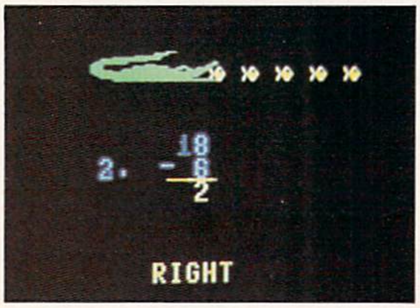



***Spellbound** requires a child to put scattered letters in order. As his skills develop, new words can be added.*

Generally, I have no sympathy for software which expects children to type LOAD "IWANNALEARN"; RUN AT 1603. On most micros, that represents pure laziness on the part of the programmer. (How about one of you hardware hackers designing a cartridge for the C-64 which will automatically boot a disk? When you turn on the computer, there could be a short program on the cartridge which tries to load a standard program from disk. If that program is not on the disk or if no drive is attached, control would return to the computer. Put the loader in a portion of memory where it is unlikely to be written over so the next disk can be loaded with the press of a function key. Software houses ought to love it.)

Since cartridge-based programs are so easy to load, they may seem to be the solution. Space limitations in cartridges, however, prohibit long or sophisticated programs. In the cartridge's 8K or 16K, the programmer must pack not only the program logic, but also text and instructions. Many programs simply cannot be implemented within those memory constraints.

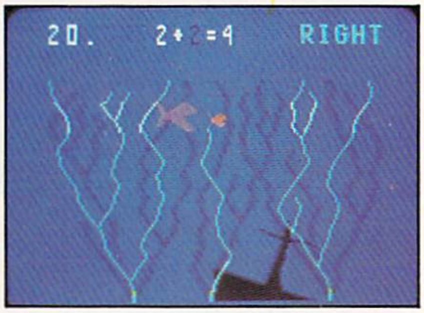
Once the program begins, the child should have the option of reading or skipping the instructions. Well-designed instructions put little actual text on any one screen and, therefore, often occupy many screens. Why should the child have to page through all that when he has used the program a dozen times? As an offsetting feature, the program should allow the child to return to the instructions at any time without having to start over. I am always amazed that kids who know programs better than I do will have momentary lapses when they truly cannot remember how to do something.




*Milliken's math programs, such as **Frenzy,** use the child's own name and encourage him to try again after a wrong answer.*

Instructions in educational software must be correct. Not just accurate for the program, mind you, but with perfect spelling and grammar. Actually, the most common place to see improper grammar is in program responses which do not follow the simple rule that single subjects require single verbs and plural subjects require plural verbs. In Island Software's *Word Factory,* the child directs groups of letters that form words into the "good" box. If a session ends with only one word in the "good" box, the screen will display "THERE ARE ONLY 1 WORD IN THE GOOD BOX." A simple routine to check for sets of answers greater than one would guarantee that as the program teaches proper spelling, it does not teach improper subject-verb agreement.

Proper grammar also includes upper and lower case, punctuation, indented paragraphs and few, if any, made-up words. For instance, the *Letters* program in *KinderComp,* which asks the child to press the key with the same letter that appears on the screen, uses both upper and lower case letters. This helps the child to understand that "J" and "j" represent the same thing. Since most home computers support lower case, educational software should use it unless there is a specific educational reason to use only upper case. *MasterType* lets you work with all upper case, all lower case, or a mix. In first learning to type, you focus on where the letters are and how to correctly position your fingers. Later, you learn to shift each letter.



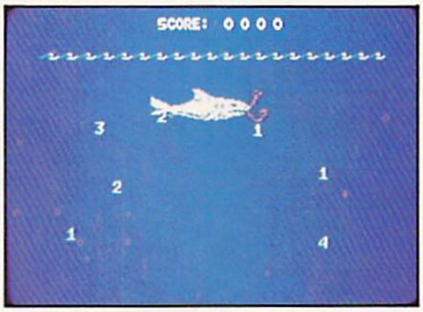

*Wrong answers to **Gulp!'s** fact and arithmetic questions propel a pursuing fish; right ones keep player's fish in lead.*

Really good instructions will use highlighting or graphics to make the important points stand out. Numerous studies, often conducted by advertisers, have shown that our eyes jump to the unusual—colorful pictures or words in italics. In *Word Factory,* groups of letters that form words are always highlighted while groups that do not remain unhighlighted. The child attends to the correct words. I also like programs which ask the child to type his name and use it to make responses more personal. Milliken Publishing Company includes this feature in its

math programs. If the child enters an incorrect answer, the program responds with "Think hard," then encourages the child to try again.

What must the child do to interact with the program? Type sentences? Control a joystick? The input device (keyboard, lightpen, etc.) should be appropriate for the child's age and developmental level. If the child must select either up or down with a joystick, the program should accept any joystick position close to the correct one. For up, it should recognize the NW, N, and NE joystick positions, not just N. Also consider your goals in relation to those of the software. If you would like your child to learn to type, keyboard input, even for a program unconcerned with typing, will help.

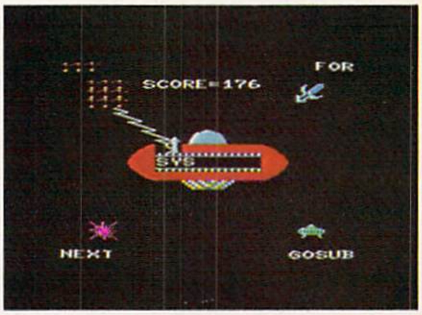
Educational software should make the best use of the computer's strengths. For computers with lousy keyboards (you know the brands), programs can let your child use a joystick or paddle. Graphics and sound are also strengths to consider. PMI's *Animated Arithmetic* contains a program named *Clown Count.* When the child correctly identifies the number of




*The* **Gulp!** *bonus game: once again, it's fish v. fish, but this time on a faster scale. From Milliken's EduFun series.*

balls on the screen, the cartoon clown will start juggling them. *Flower Power Math Fun* from Softwave Productions puts arithmetic problems on a colorful background of flowers, sky, and flying birds. Other than this, the program uses no flashy graphics displays. This, the company claims, prevents children from being distracted.

Why shouldn't the software make use of all the computer's color and sound capability instead of being a mute black-and-white special? Color and sound need not distract from education, but can actually enhance it. Well-designed programs will use these strengths of home computers to draw your child's attention to the learning problem. But beware these features when they are unrelated to an educational purpose. While the child decides on an answer in *KinderComp,* the title on the screen changes color one letter at a time. Undoubtedly, the program looks good running as a demo at your local computer store, but such distractions will ultimately impede




**Master Type** *by Broderbund overwhelms you with attacking spaceships represented by letters which you must type.*

your child's concentration. The purpose of color, sound, and graphics should be to capture the child's interest, then hold it long enough for him to learn.

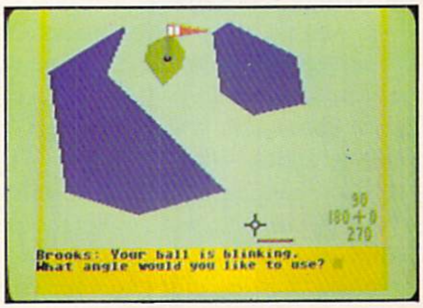
## *ERROR HANDLING & WARRANTY*

No educational program which can be "bombed" by an incorrect response is designed well enough for your child. The best programs will even ignore the break and reset keys. At the very least, the program should start over, not just return to the computer's operating system with some cryptic error message. Load the program and intentionally hit the wrong key, trying to cause an error. If you can, your child surely will. Even if you cannot, who knows what your young computer whiz can do.

Children, especially boys, like to experiment with programs outside the bounds of rules and instructions. When you do something wrong, and the program does not crash, what does it do? If a funny monster walks across the screen carrying a banner that says "Now don't do that!" to the tune of "101 Trombones" in nine part harmony, do not expect your child to learn from the program. He will certainly play with it a lot, but probably only by giving wrong answers.

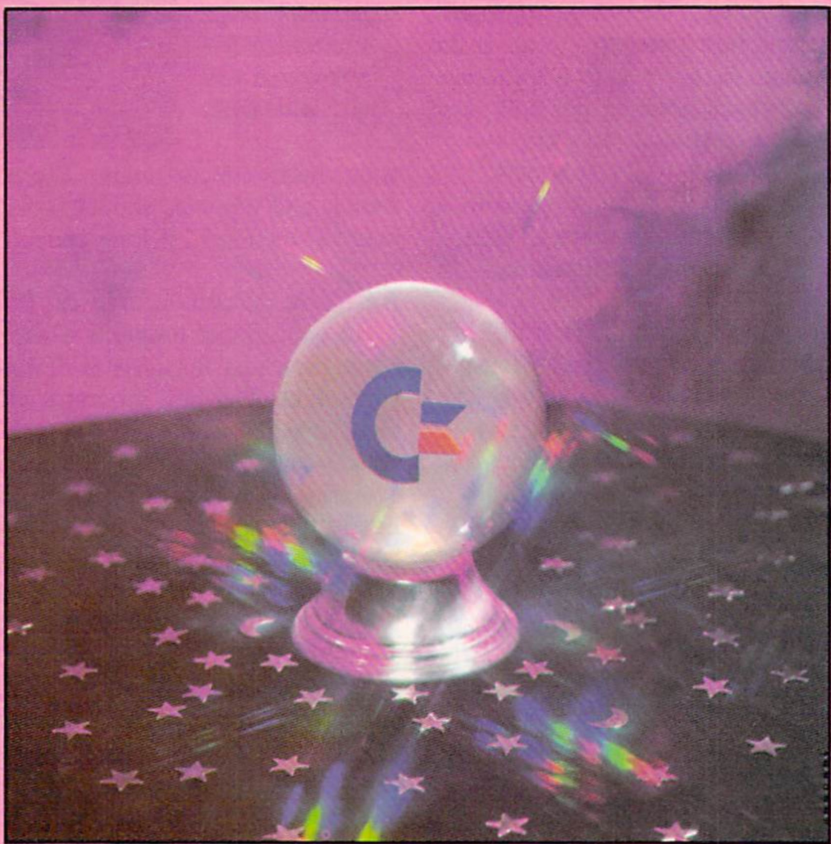
Does the program allow you to give clearly incorrect answers? If you need to type a number, will it accept a letter and then tell you your answer was wrong? Or will it just sit patiently, without response, waiting for you to type a number? Remember a keyboard itself is a strange beast to your child and will be the cause of many typing errors.

*Continued on page 71*




**Golf Classic,** *for grades 4 up, practices angle and length estimation and geometric skills. For one to four players.*

# THE FUTURE OF COMMODORE?



© 1984 Ross M. Horowitz

*By Sheldon Leemon*

The explosive growth rate of the microcomputer industry in the last five years has naturally created some turmoil for its participants. But for Commodore International Ltd., turbulence has always seemed to be more the exception than the rule. Take, for example, the events of this January. On January 7, at the Winter Consumer Electronics Show, Commodore was the only major manufacturer of home computers in the under-$500 market to announce a new product line. The new Commodore 264 and 364V computers which they introduced featured built-in applications software on ROM, such as word processing and spreadsheet programs, which allow the user to work with the computer as soon as he takes it out of the box. The company also announced that its 25th anniversary year was one of record profits and sales, in a business where major corporations such as Texas Instruments, Mattel, and Atari have been left licking their wounds. Yet on January 13, Jack Tramiel, Commodore's founder and chief executive, announced his resignation. Shortly thereafter, four other executives also left, including Donald Richard, acting president of Commodore U.S., and it was announced that the 264 would not appear in April as had been previously announced. Wall Street reacted quickly and sharply to the news, sending Commodore stock plunging from $47 a share to just $28 within a three-week span.

Should the stock market's concern be shared by the average Commodore computer owner? Let's examine the impact that recent events will have on the management of Commodore International, its products, and its customers, and then try to assess what

the future may bring for Commodore.

## *JACK HITS THE ROAD*

The event that overshadows almost all others is the resignation of Jack Tramiel. The uncertainty left in the wake of this resignation is understandable; for the company's entire 25-year history, Mr. Tramiel has been the very heart and soul of Commodore. He infused the organization with his personal philosophy of business, characterized by aphorisms such as "Commodore builds for the masses, not the classes," and "The minute you're through changing, you're through." His obsession with controlling the cost of every phase of the manufacturing process led to Commodore's acquisition of MOS Technology and other semiconductor manufacturers, so that the company's manufacturing facilities around the world now produce almost every part needed for every Commodore product. Along with other strict cost control measures imposed by Tramiel, Commodore's vertical integration has allowed it to earn record profits in the midst of a price war which has left competitors reeling from huge losses, driving some out of the computer business completely.

Although overall the company has thrived on Tramiel's relentless, hard-driving style of management, his strong-willed personality also caused its share of problems. He had never been able to establish very good relations with computer dealers, despite the frequent restructuring of Commodore's U.S. marketing organization in recent years. Under his reign, computer retailers have accused Commodore of treating them as harshly as if they were suppliers or competitors, and as a result, many have become disenchanted with Commodore and dropped the product line. And there has been a steady turnover of top executives at Commodore as a result of Tramiel's inflexible one-man rule.

Now that Commodore's sales have passed the billion dollar mark for 1983, it has become increasingly clear that the company is just too big for one man, however talented, to run. Tramiel himself said after his resignation, "I'm not a professional executive. I use a personal touch, like a family business. I never believed I would be able to run a business of that size." Commodore's new president and chief executive, Marshall F. Smith, on the other hand, is an experienced professional manager. As president of Thyssen-Bornemisza Inc., a diversified manufacturing company and U.S. unit of one of Europe's biggest manufacturing concerns, Mr. Smith has directed the activities of 10,000 employees in 23 subsidiaries. Commodore chairman Irving Gould is relying on Smith's experience in manufacturing, marketing and finance to strengthen Commodore's position as a major international corporation. So while Commodore is losing its flamboyant founder, it may now have the opportunity to enjoy a period of more stable growth.

When I asked Neal Harris, head of Commodore's publications, what changes he thought Tramiel's departure would have on the type of computers Commodore manufactured, he replied that he expected no radical change from current plans to concentrate about 50% of its efforts on the under-$500 product category, 30% on the $500-$1000 range, and 20% on the over-$1000 price class. Let's take a closer look at current and future activity in each of these three categories.

## *DON'T TAMPER WITH IT*

It is in the under-$500 price class that Commodore has had its greatest success, growing from $50 million in annual sales a few years ago to over a billion dollars in sales last year on the strength of the VIC 20 and 64. Those 64 owners who have been distressed by rumors that the 64 was going to be replaced by a newer model such as the 264 will be greatly reassured by the statement that Commodore's Chairman Irving Gould made in late January concerning the company's current and future product plans. He stated that the 64 was "by far the best selling microcomputer in the world" and that demand for it "still continues to exceed our production capabilities even during this post-Christmas quarter." As a result, he said, Commodore is continuing to increase its 64 production capacity, and will not even introduce the 264 until sometime later this year "when our capacity permits both the continuing increase in 64 production and large scale production of the Commodore 264." Clearly, the recent retreat of Texas Instruments and Mattel from the under-$500 computer market has lessened competitive pressures to rush the introduction of more powerful new computers at lower prices. And IBM's introduction of the PCjr, a machine with relatively limited technology at a relatively high price, not only presented much less of a threat than the industry had feared, but actually encouraged competitors such as Atari and Coleco to raise the prices of their home computers. This stabilization of the low-end computer market leaves the Commodore 64 almost in a class by itself, greatly extending its popularity, and ultimately its life-cycle. Owners of the 64 can rest assured that increasing sales and production will mean increased software and peripheral support both from Commodore and from independent manufacturers.

## *THE DREAM MACHINE?*

But what of the 264? As introduced at the Winter Consumer Electronics show in January, its place in the current product line is exceedingly hard to define. If it is to be sold as an under-$500 computer, it would have to be measured against the strongest competitor in that class: Commodore's own best seller, the 64. On the one hand, the 264 lacks the 64's superb sprite graphics and three-part music synthesi-



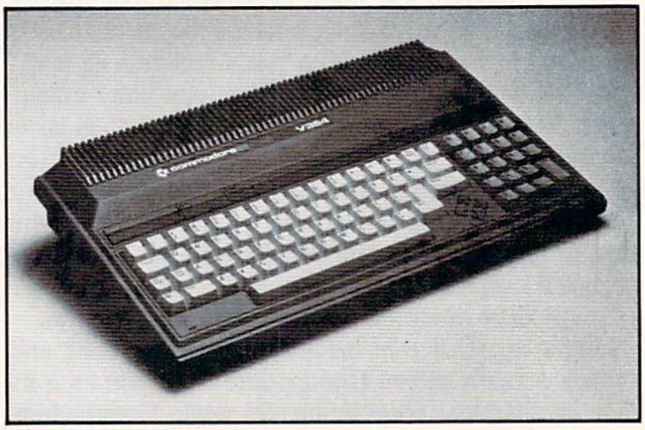
*DPS 1101 printer, 264 computer, and SFS disk drive.*
*READER SERVICE NO. 189*

zer. Ont he other hand, it has the same 64K of memory, plus a greatly expanded operating system. With its extended BASIC, built-in extended machine language monitor, enhanced screen-editing capabilities, and room for 32K of built-in applications software such as a word processor or spreadsheet program, it is from a software standpoint almost a "dream machine" for Commodore users. The tradeoffs between enhanced software and reduced hardware capabilities of this machine make it very similar in its appeal to the 64. The 264 had been rumored to be somewhat more expensive, in the $350-$400 price range (no official price was set by Commodore at the time of its announcement). But price alone obviously can not be used as the differentiating factor. If the new machine is priced far above the 64, it may be rejected by consumers, while if it is priced too closely to the 64, it might compete directly against it, and simply shift sales away from the 64 without increasing Commodore's overall sales base. Retailers might balk at having to stock hardware and software for such similar, yet incompatible, products. The continued strength of the 64 past the big Christmas sales period has obviously given Commodore a much-needed opportunity to more carefully assess its marketing strategy for the 264 before bringing it out.

Commodore insiders have been hinting that when it finally does come out late in 1984, there is a good possibility that the 264 will be sold at least initially as a 16K Commodore 216. Such a machine was also shown at the January CES, although with a great deal less fanfare than the 264 and V364. This configuration would correspond more closely to the original design concept of the machine, which started out as a low-cost replacement for the VIC 20. As such, the computer could sell in the $100 price range, where it would be clearly differentiated from the 64 by its lack of sprite graphics and SID-chip sound, as well as its smaller size, smaller memory, and lower price. The VIC 20 has been declining in sales since the 64 reached its peak of popularity, and introducing a 216 to replace it would inject new vigor and excitement into the lowest-price segment of the computer market, while allowing Commodore to retain a hold on the shelf space which retailers currently have allocated to the VIC.

And what a $100 computer the Commodore 216 would be! Since Commodore has already spent a considerable amount of time and money developing software to work with the Operating System and TED chip used by the computers that were shown in January, it is safe to assume that regardless of the final configuration of the new under-$500 computer, it will have the features associated with that combination of hardware and software. The first of these features is advanced screen editing. The new keyboard has separate arrow-shaped cursor movement keys clustered in a diamond pattern for easier cursor movement. An Escape key has been added, and using certain escape sequences performs editing functions similar to those provided by word processors. You can erase text from the current position either back to the beginning of the line, or forward to the end of the line, cancel quote made or insert mode, and set an auto-insert mode in which any text inserted will push the rest of the line forward as you type, rather than writing over existing text. By pressing two keys, you can scroll the screen up or down, insert a blank line or delete a whole line, and even set top left and bottom right margins for creating windows.

The BASIC 3.5 included in this new Operating System goes far beyond the BASIC 2.0 which is found on the Commodore 64. It adds almost 50 new commands. For program editing there are AUTO line-numbering, RENUMBER, and block DELETE commands as well as TRON and TROFF for tracing, a HELP command that highlights an illegal statement in inverse characters, and KEY, which allows you to define a string of characters which can be printed out by striking a single function key. Almost all of the BASIC 4.0 disk commands, such as DIRECTORY, DSAVE, DLOAD, HEADER,



*The V364 adds voice synthesis, more built-in software.*
*READER SERVICE NO. 190*

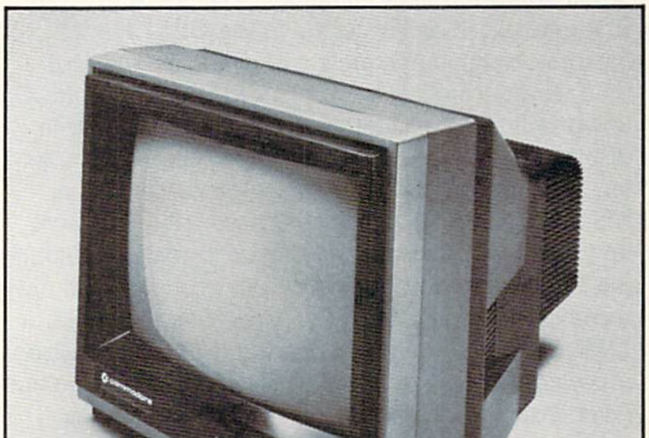


*C1531 cassette drive, C-64 with VICModem, MPS 1101. READER SERVICE NO. 191*

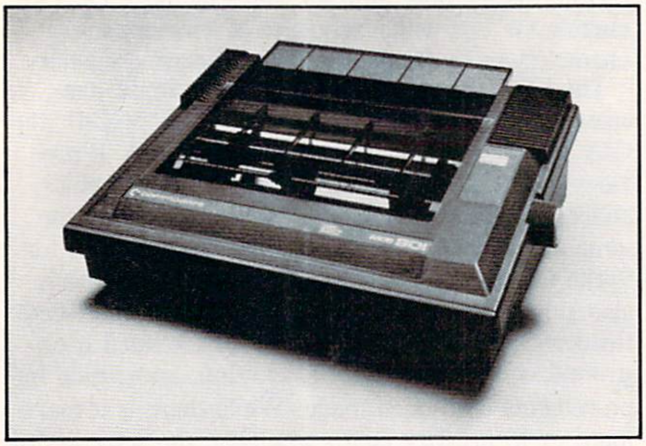
SCRATCH, COLLECT, COPY, RENAME, and BACKUP are included, as is the faster string-collection method used by BASIC 4.0. Structured programming is supported with commands such as DO, LOOP, EXIT, WHILE, UNTIL, and IF-THEN-ELSE. TRAP and RESUME are provided for error-trapping, along with reserved variables ER, which contains the error number of the last error, EL, which gives the line number at which the error occurred, and the ERR$ function, which gives the error message that would be generated by a particular error. For the first time, this new Commodore BASIC fully supports the graphics and sound features of the machine with commands like VOL, SOUND, GRAPHIC, DRAW, PAINT, BOX, CIRCLE, LOCATE, CHAR, SCALE, GSHAPE, SSHAPE, SCALE, COL, RGR, RCLR, RLUM, and RDOT. These commands allow the user to easily control high-resolution or multicolor bit-map graphics in either full screen mode or split screen mode, with five lines of text at the bottom. As an example of how powerful these commands are, CIRCLE can take up to nine different parameters—color, X and Y coordinates of the center, horizontal radius, vertical radius, starting arc degree, ending arc degree, clockwise rotation and degrees between segments. This makes it possible to draw any kind of circle or ellipse, even geometric figures like triangles, pentagons, or hexagons, in whole or part, in any proportion or rotation with just one BASIC statement. This represents quite an improvement over the PEEKs and POKEs required for VIC and 64 graphics from BASIC!

The new Operating System also has a built-in machine language monitor. It has all of the commands found on the Tiny Monitor featured in the PET/CBM series, such as M for displaying and changing memory, G for executing machine code, R for displaying and changing registers, S for binary save of memory, L for binary file load, and X for exit to BASIC. But the new monitor also includes some commands usually found only in extended monitor programs, such as an A command for simple assembly, a C for comparing memory segments, a D for disassembly, an F for filling memory, an H for hunting for a particular sequence of numbers or text characters and a T command for transferring memory.

The TED chip is a new integrated circuit that takes over most of the functions that the VIC-II, the CIA, and the SID chips perform in the 64. It provides 320 X 200 high-resolution bit map graphics, and 160 X 200 multicolor bit map graphics in the same format as on the 64, but has no provision for Sprite Graphics. The character graphics scheme is also similar to that of the 64, allowing for user-defined character graphics, as well as extended background color text mode. In addition, some nice improvements have been made. While there is a maximum of 16 primary colors available, the selection of colors is better (3 of the 64's colors are shades of gray), and each color except for black can be set to one of 8 different luminance levels, for a total of 121 possible shades. This makes a striking difference in the display quality, allowing many more combinations of colors to appear on the screen at once without the "smearing" that occurs on the 64. Another addition is that any test character can be designated as a "flashing" character, which will blink on and off like the cursor. The TED chip has two simple tone generators, one of which can also generate "white noise" for sound effects. Other hardware improvements include a faster internal clock, which allows the processor to execute instructions about 75% more quickly than that of the 64. And while memory expansion is limited to 64K of RAM, the new system uses a more sophisticated method for bank-selection of ROM programs that allows it to chose from any one of four 32K ROM programs at a time. The Operating System also provides for using bank-selection with BASIC, so that if a full 64K of RAM is present, the BASIC interpreter can use about 59K



*C1703 color monitor works with VIC, 64, 2/364, SX64. READER SERVICE NO. 192*



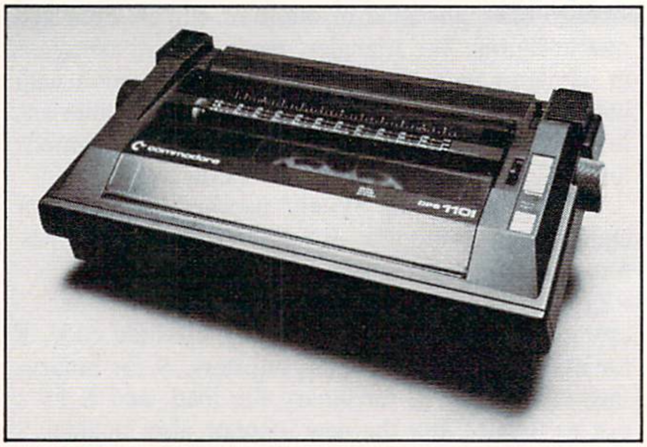
***MPS 801: a unidirectional color dot matrix printer.***
***READER SERVICE NO. 193***

for program space. While the 264 model introduced at the January CES show was to have come with a 32K ROM program for word-processing or business calcualtion built in, it is not known for certain at this time whether this feature will be standard or optional on the final model, and no decision has been made as to which software titles might be included.

## *CONTROLLING THE MIDDLE*

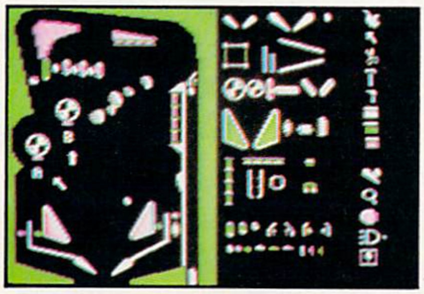
In the $500-$1000 price range, two of Commodore's long-awaited products are finally coming to market. The first is the SX-64 (also known as the SX-100 or Executive 64), a suitcase-portable version of the 64, with a beautiful detachable keyboard, a built-in 1541 disk drive, and a 5" color monitor. At a selling price of less than $1000, this unit has generated a lot of interest, and since demand is already so high for the regular model of the 64, it may be in short supply for quite some time. The other product is the B-128 business computer. The 'B' series was first announced as successor to the CBM line of business computers almost two years ago. At that time, several models were planned, ranging from a low-profile 128K model, without the usual built-in display screen, to a 256K all-in-one unit with monitor, disk drives, and an 8088 co-processor to run MS-DOS (IBM PC-compatible) software. For now, however, all of the more elaborate models, including the PC compatible, have been put on hold. The only member of the 'B' family to emerge is the B-128, a 128K low-profile model using the 8-bit 6509 microprocessor, priced somewhere under $1000. This computer displays 25 rows of 80 column text on the user's external monochrome monitor, has a sculptured keyboard with programmable function keys and numeric keypad, and an advanced BASIC version 4.0+. Most of the advanced editing, monitor, and BASIC features discussed above in connection with the 264 (except for graphics) apply to the 'B' as well. One rather odd feature is that a cassette port is included, but since the Operating System has no cassette support built in, you would have to load a cassette driver from a disk in order to use it! The more functional I/O ports include a built-in RS-232 interface, as well as the familiar IEEE port which allows the use of Commodore's standard line of printers and floppy and hard disks. Ed Havlik, sales manager for Commodore's system group, says that although the company's business line of computers has in the past gained wider acceptance abroad than in the U.S., he is planing an aggressive campaign to market the new 'B' series here. This campaign includes offering the new computers as part of a total system package, at a much lower price than that ordinarily charged for the individual components. One such package will include the B-128, the 8050 dual disk drive, and a dot-matrix printer, while another puts together a B-128, 8050 disk drive, and 6400 letter-quality printer with word-processing, data base, and accounting software. "That's everything that a small business could need," says Havlik. Since Commodore does not set suggested retail prices on these products, Havlik could not give exact figures for these packages, but he did say that "the savings will be substantial, and the prices will be very, very attractive." He went on to say that because of his background as both sales representative and a computer retailer, his selection to head up the sales effort on the B-128 demonstrated Commodore's commitment to support its full-service dealers, and to expand the number of full-service Commodore outlets. The number of such outlets has declined somewhat in recent years, as many dealers have felt some resentment at their inability to share in the success of Comodore's low-end machines, which for the most part have been sold by discount and chain stores. The support of computer retailers is vital if Commodore is to be at all competitive in the higher-end personal and business computer market. So if Commodore's new campaign proves a success, you could see the more advanced 'B' models appear very quickly.

*Continued on page 74*



***DPS 1101: bidirectional impact daisy wheel printer. It and the MPS 801 are scheduled for year-end release.***
***READER SERVICE NO. 194***



*READER SERVICE NO. 140*

## *PINBALL CONSTRUCTION SET*
***Electronic Arts***
***Commodore 64; Disk***

Often, no matter how fantastic a computer game is, there are one or two elements that you wish were slightly different. Even with multiple difficulty levels and play options, the game could still stand a few changes. Maybe the graphics and sound need slight alterations, or perhaps you'd like to make the scoring easier or harder, or speed up or slow down the action.

Luckily, the trend in today's home entertainment software is towards games that allow the user to adjust many, in some cases all, of the parameters of the game, then save the new versions on disk. One game on the cutting edge of of this trend is *Pinball Construction Set* by Bill Budge.

As the name implies, *PCS* lets you constuct your own pinball tables from scratch and then further personalize them with your own custom paint job. In addition, you choose the weight of the ball, its speed, the amount of force the bumpers "kick" with, and the bounce of the. flippers and solid parts of the table.

All this is accomplished by using a joystick-controlled hand to drag flippers, bumpers, spinners, rollovers, etc. from the Parts Box onto the table. Beside the usual assortment of pinball parts, the Parts Box also contains "polygons." These are shapes which can be brought onto the table, painted, and then, by using the hammer, scissors, and arrow icons, stretched into any shape to make up the solid parts of the table.

Other icons include a magnifier, world, AND gate, and disk. The magnifier is used with the paint brush to paint fine details onto any of the game's parts or to design the game's back glass which appears above the scores. The world icon lets you adjust gravity, speed, bounce, and kick.

When you select the AND gate, you are presented with two more icons—a screwdriver and a pair of pliers. These are used to wire various elements of the table together to create bonuses or activate the bonus multiplier.

For example: let's suppose the pinball game you've created features a bank of drop targets on each side of the table. By wiring the two banks of drop targets together, you can give yourself a bonus of, say, 10,000 points each time you knock down all eight (four on each bank) targets. Just like many real tables, the computer remembers how many and which targets you've already knocked down, even if there are four people playing.

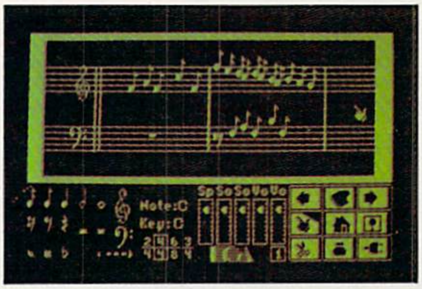
Once you've created your "dream table," you can save the game by using the disk icon. One of the things which really puts *PCS* ahead of similar efforts is that while most games that let you create your own playfields require that you first load the main program before running your own version, *PCS* allows you to create a totally independent piece of software. This means that you can design a table and give the finished disk to another C-64 owner even if he doesn't have *PCS.*

Whether you're a flipper fanatic, or just a fan of great computer games that let you do some of the creating, you owe it to yourself to pick up *Pinball Construction Set*. It is among the best home entertainment programs ever written.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403.

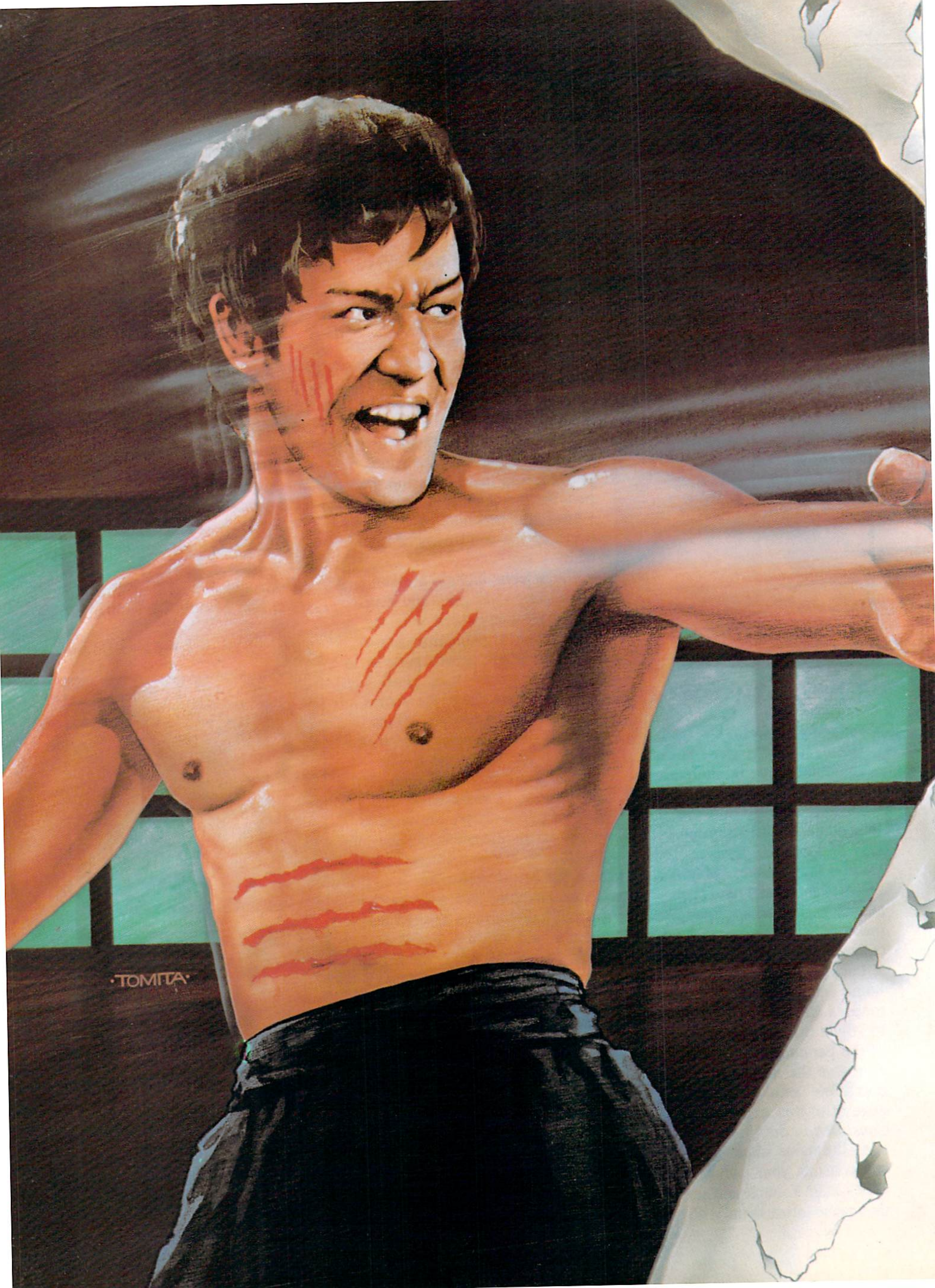
*—Lloyd Davies*



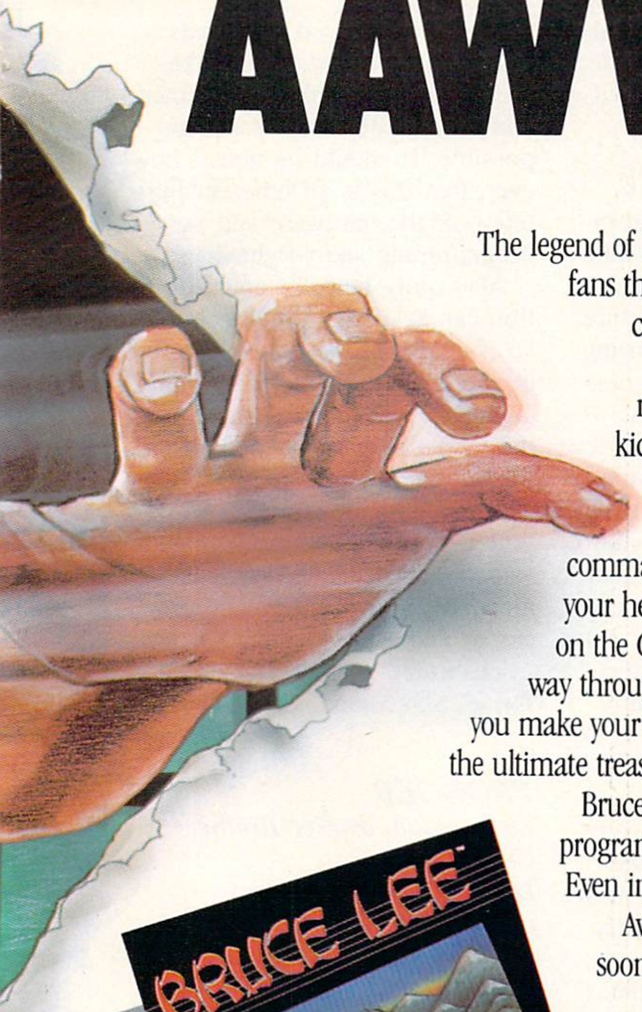



*READER SERVICE NO. 141*

## *MUSIC CONSTRUCTION SET*
***Electronic Arts***
***Commodore 64; Disk***


·TOMITA·

# AAWWIIIIEEE!



The legend of Bruce Lee lives on in the imagination and memory of millions of fans throughout the world. And now, through the awesome power of the computer, you too can relive the power of Bruce Lee.

Bruce Lee™ combines state-of-the-art technology with the masterful moves of the martial arts. From devastatingly lethal kicks and staccato thrusts to the unrestrained fury of every reflex, the Bruce Lee game explodes with action.

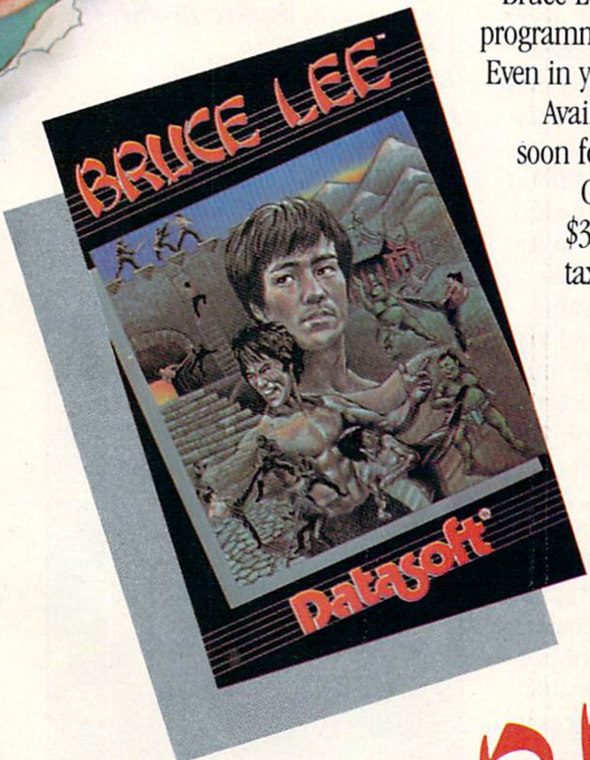
All the force and controlled discipline of Bruce Lee is at your command. You'll see it in his smooth and graceful leaps. You'll feel it in your heart as you prepare to do battle with his ominous adversaries. Take on the Green Yamo and Ninja in hand to hand combat as you fight your way through 20 mysterious oriental settings. Unexpected dangers loom as you make your way past exploding bushes and the fire wizard in your quest for the ultimate treasure.

Bruce Lee. It not only lives up to the expectations of the best programmers and players around. It lets you live out the life of a legend. Even in your spare time.

Available now for Atari and Commodore 64 Computers, coming soon for the Apple II series and IBM PC and PC/JR.

Contact your local dealer or send check or money order with $3.00 postage and handling. California residents add 6.5% sales tax to Datasoft.

Suggested retail $34.95.






# BRUCE LEE™

By
**Datasoft®**

Datasoft, Inc. 19808 Nordhoff Place, Chatsworth, CA 91311

If you have ever tried to write your own music, you know it can be a lot of hard work. If you don't have a piano or other suitable instrument handy, you have to go strictly by theory without a chance to hear your ideas expressed. Those who usually sit at the piano and improvise until they hit upon something have to laboriously transcribe what came out of their hands onto staff paper, note by note.

Will Harvey's *Music Construction Set* and your C-64 will solve these problems for you and take the drudgery out of composition forever. Not only does this program let you quickly and easily move notes around the staffs with a joystick or a KoalaPad, you can immediately hear your composition interpreted by one of the best performers in electronic music—your C-64's SID chip.

When you first boot the disk, you can sit back and listen to thirteen preprogrammed compositions that range from classical to fifties swing. If you've ever doubted your computer's musical capabilities, these selections will make a believer out of you.

Once you've been sufficiently inspired and decide you'd like to write some music, grab your joystick. Now, by controlling the on-screen hand, you can choose a time signature and begin dragging notes up to the staffs at the top of the screen. Although the staffs are arranged in standard fashion with a treble clef at the top and a bass clef on the bottom, you can change this simply by pulling off, say, the bass clef and replacing it by a second treble clef. You can even switch back in mid-composition.

Below the staffs to the left is the assortment of notes and rests at your disposal. These range from whole notes to 16ths. 32nd notes are even available, but only through the keyboard. There are ties and dots which can be used to affect a note's duration.

In the same area you can choose from a sharp or flat to raise or lower the pitch of an individual note by a half step, and a natural to temporarily cancel a sharp or flat in the key signature. The key itself is chosen by using the hand to point to the key indicator and pressing the button.

Also at the lower portion of the screen are graphic representations of five slider type controls. By using the hand to move the sliders up and down, you can adjust the tempo, tonal quality of each staff individually, and overall volume of both staffs. These affect not only compositions currently under construction, but also complete works that have already been saved on disk, and then loaded. Therefore, you can experiment with the preprogrammed selections to discover exactly how each slider works before you try to write your own music.

Finally, we come to the nine icons at the lower right. These are the heart of *MCS* and, to a large extent, what put it miles ahead of most music programs.

By pointing to the left and right arrows and pressing the fire button, you cause the staffs to scroll in the desired direction. Pointing at the piano will allow you to hear whatever's already on the staff. The small house is used to return you to the beginning of composition, and the plug icon lets you pick joystick or KoalaPad as your main method of control.

If you get the impression that *MCS* is one of the best programs available to help serious musicians compose, you're right. As great as this program is, however, it does have a few drawbacks.

First, you can only use three voices at a time, so full triads plus bass are impossible. Likewise, full 7ths, 13ths, augmented, and other exotic chords are impossible. It should be noted, however, that this is an inherent limitation of the hardware and not programming short-sightedness.

Also, only notes of equal duration can be used to make chords so if you want the equivalent of a quarter note at G sounding simultaneously with a whole note at C the whole note has to be constructed of four tied quarters.

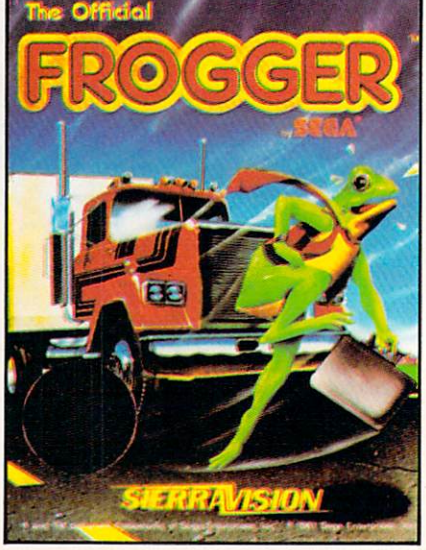
These limitations aside, *MCS* will aid both experienced songwriters and dedicated novices alike. It's a powerful music processor and a joy to use.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403.

*—Lloyd Davies*

*FROGGER*
***Sierravision, Parker Brothers***
***C-64***
***Disk, cassette, cartridge; joystick, keyboard***

In our never-ending search for industry trivia, we tried to find out from Sega why the rights to the computer version of *Frogger* were split; Sierravision has re-




*READER SERVICE NO. 142*

leased the disk and cassette versions and Parker Brothers has released the cartridge. No one at the concerned companies could answer the question as of press time, but I'd be glad to speculate: Parker Brothers had first rights but they only have the facilities to produce cartridges. Sega insisted that disk and cassette versions be released as well, and *voila,* Sierra stepped in. Is there bad blood between Sega and Parker Brothers? Who knows? Who cares? Let's play *Frogger.*

The game is probably familiar to most of you. The gamefield is an overhead view of a four-lane highway, a river and its two banks. The player must maneuver a frog from screen bottom to screen top across the highway (without getting squashed by its heavy traffic) and across the river by hopping turtles and logs, which are moving horizontally with the current. Complicating matters are snapping crocodiles and turtles which will abruptly submerge. A lady frog occasionally appears and if you, ahem, jump her and arrive home safely, you win extra points. Home is a series of cubicles at screen top, some of which are empty, some of which have an insect (bonus points), some of which have a crocodile (froggy death). When all of your frogs are safely across and home, it's on to the next, more difficult screen.

Both the disk and cartridge versions of *Frogger* are excellent, little to choose between them. (We have not seen the cassette version.) The disk's graphics were a little more crisp and dimensional. The disk also allows the option of keyboard control. The *Frogger* theme plays throughout the disk version's gameplay, but only in the introduction of the cart version. The controls in both versions are sure, and both have fast and slow gameplay options.

*Frogger* was all the rage last year, and for good reason: it's a fun game to play for all ages. It remains to be seen if it was last year's game, period, or whether it will become a perennial favorite.

Parker Brothers, 50 Durham Road, Beverly, MA 01915.

Sierravision, Sierra On-Line Building, Coursegold, CA 93614.

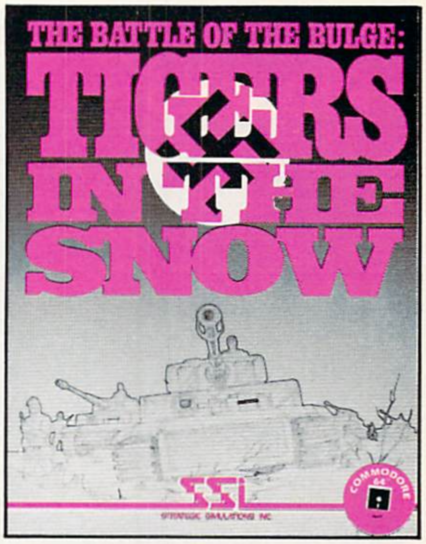
*—Tim Moriarty*

## *TIGERS IN THE SNOW*

***Stragetic Simulations, Inc.***
***C-64***
***Disk; keyboard***

Playing this game is like stepping into a time machine and being transported back to World War II's greatest battle: The Battle of the Bulge. A complex, serious game, *Tigers in the Snow* is only for those willing to pore over the instruction booklet: four 8½ × 11" pages crammed with type. It's also only for those willing to forego arcade action: this is a board game on a computer screen, with less movement than you'd get from jostling your *Scrabble* table. Which is a shame, because if the graphics of the same company's *Combat Leader* could be utilized, this one would be unbeatable. However, what *Tigers in the Snow* lacks in moveability, it makes up in realism.

The game is divided into several phases. The Hun goes first, running through the Movement Phase and seven-part Combat Phase (including unit selection, artillery, strategy, retreat/advance, etc.) The Allied player then enters his moves. This process occurs twelve times (once for each day from December 16 to December 27, 1944). Following the Reinforcement Phase, the Victory Status Phase reports the winner. The game closely approximates




*READER SERVICE NO. 143*

the real battle. As in WWII, the Allies will try to cut off the German supplies. Weather is also a key factor, because an overcast day can hamper your air support.

Two charts—one supplying a quick reference to game rules, the other a map of the TV screen playfield and symbol meanings—will help you to keep track of what's going on. So does the computer, with constant updates on your forces, the terrain they're in, strength, supplies, and the like.

During the last few turns, you'll be given hints about who the winning army will be. This gives you a chance to go for broke. You can also save a game in progress to disk or cassette if you're too weary to complete it n one sitting. War, after all, is hell.

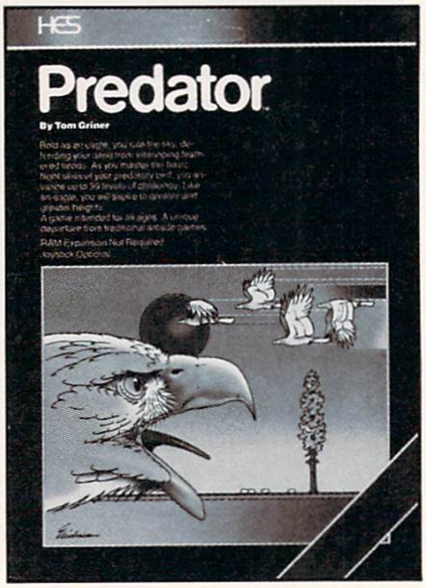
Strategic Simulations, 883 Stierlin Rd., Mountain View, CA 94043

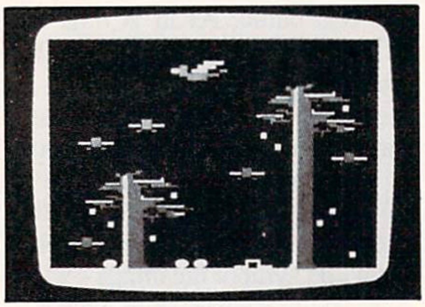
*—Lenny Nelson*

## *PREDATOR*

***HesWare***
***VIC 20***
***Cartridge; joystick, keyboard***

Give HesWare credit. Not content to create variations of existing games, the company has chosen the far more hazardous



*READER SERVICE NO. 144*

course of developing all-new ideas . . .with sometimes uneven results. Such as the case with *Predator.*

The game begins with a large, ungainly bird dropping an egg onto the playfield. This egg bursts into many smaller eggs, from which hatch smaller birds. You, as the predator, must either pick up the eggs prior to their hatching or shoot down the hatched birds. Impeding your efforts is a pair of trees through which you are unable to fly. To reach eggs outside the confines of the trees, you must fly above them.

In order to fly, you must constantly move the joystick up and down. Should you stop, gravity takes over and you plummet to the ground.

One further obstacle to success are worms which must be devoured before they reach unhatched eggs. When worm comes in contact with egg, a mutant bird results. These are extremely deadly to your predator.

Gameplay is just that simple, which is not to imply that it is easy. There are 99 skill levels. *Predator* does boast some colorful, ticklish graphics. When you lose a predator, it is plucked to death by a rival bird amid a flurry of feathers. It would be amusing to watch if it were not for the grisly scenario. My favorite is the achievement of high score. The screen blanks except for a tiny rocket and a little man. The man ignites the rocket and, at apogee, the rocket explodes into blue sparklers that proclaim you as high scorer, master predator.

Hesware, 71 Park Lane, Brisbane, CA 94005.

*—R.J. Michaels*

## *THE COMPUTER MECHANIC*

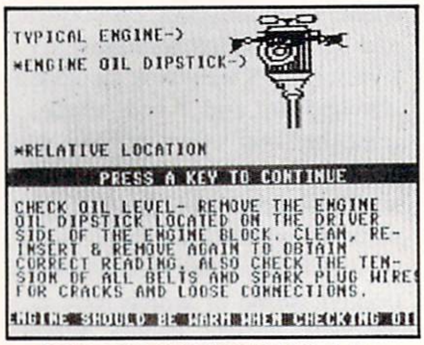
***Softsync, Inc.***
***C-64***
***Disk, cassette***

*The Computer Mechanic* is, in concept, a tremendous program for keeping your car on the road and reducing repair costs. It includes sections on how to maintain your vehicle's major systems and when to check them. If your car won't start or runs rough, this software will lead you gently through a check of the most likely problems. You can even store a tuneup record of sorts.

As an instructor, the program rates fair. In a series of seven screens, it pictures major systems: brakes, suspension, tires, fuel and emissions, transmission, engine, and cooling. Simple pictures of each are shown and instructions on simple maintenance procedures are given. Variations among different cars are not indicated. (You'll never know that the transmission of your Alfa Romeo Alfetta is in the rear end).

A similar set of seven screens lists the frequency, at which you should maintain twenty-seven different parts of your car. If you have loaded the maintenance record of a particular vehicle, the mileage at which it is due for a check or parts replacement is specifically provided.

On that bright sunny morning when you're headed to the beach but your car isn't, *The Computer Mechanic* can give you valuable advice on what to look at. Based on how you answer questions about your car's problem, the program leads you through detailed checks of the battery and spark plugs as well as suggesting that you look at such things as the air filter, cables, and distributor. The questions here should really help you zero in on the problem.




*READER SERVICE NO. 145*

Tuneup records for up to 100 vehicles can be stored. (Why anyone with 100 cars would need the simple instructions in the rest of the program, I don't know.) The entries allowed—oil change, tuneup, brake job, tire rotation, and transmission check—are not detailed enough for a meaningful record. And since the record only keeps the last date and mileage of each vehicle's maintenance, you cannot keep a running historical record. The only way around this

is to enter a new record, under a different name for your car, after each tuneup.

*The Computer Mechanic* is a grand idea; too grand for a single-load BASIC program. Recordkeeping is inadequate. Tuneup instructions are general and won't fit some cars. The section on diagnosing a dead auto is the only really valuable one. Also, please use jack stands (not just the jack) when you crawl under you car. The program's only advice is to use "normal safety precautions."

Softsyne, Inc. 14 East 34th Street, New York, NY 10016

***—Richard Herring***

## *KIDS ON KEYS*

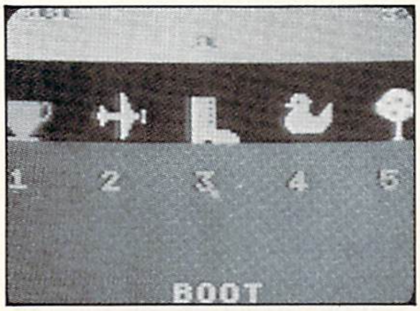
***Spinnaker Software***
***C-64***
***Cartridge***

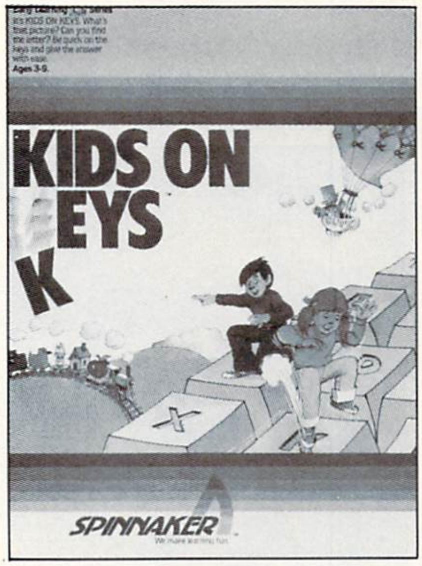
Recently my daughter's teacher was describing the computer program in her school. "The only problem is," she said, "that children don't know the computer keyboard because they usually don't take typing until high school." With increasing computer education at the elementary level, she added, "We're just going to have to start introducing typing skills at a much earlier age."

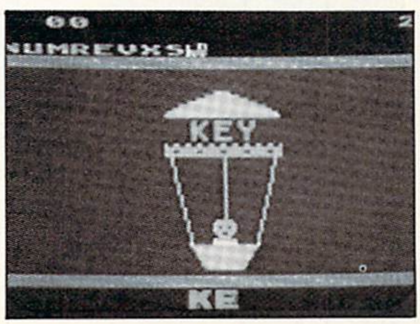
*Kids on Keys* by Spinnaker is a positive step in that direction. Here is a program that combines letters, numbers, words, and typing in three different games that children from 3 to 9 will find absolutely enchanting. Simultaneously, parents and teachers will applaud its practicality.

In the first game 15 letters/numbers float down from the top of the screen. The child must type the corresponding letter on the keyboard. At the end of each of the four rounds in Game One a balloon descends, displaying words which the child has to type correctly. Then, with musical fanfare, the balloon ascends and another round begins. In each succeeding round the speed of the game increases, and concurrently (hopefully!) the little fingers typing become that much more proficient.

Game Two has a slightly different twist. In this game five different pictures float down the screen, and the child types the correct word in a timed period. At level 1 only the first letter of the word is required (i.e. "f" for flower) but in levels 2, 3, and 4 he has to type the entire word before it disappears from the screen.




READERS SERVICE NO. 175

With Game Three there are five pictures shown at the top of the monitor, each numbered 1, 2, 3, 4 and 5. As one of the pictures drifts down the screen, the child types the correct number (shown below) that identifies it before the word fades off the screen.

Throughout Game One, Two, and Three the child earns points for correct answers. And they love it! After a particularly strong round the computer will announce "new high score." They eat this up!

*Kids on Keys* is a good program. The color and graphics are delightful. The musical background and sound effects are really quite clever. And the program is as easy to work as, well, A B and C!

Does it get 5 stars? No way, and here's why. At the end of each round in Games Two and Three there is a Bonus Round. Holy cow, if this is Spinnaker's idea of a Bonus, I'd hate to see their Penalty Round. Because here only a partial picture floats down the screen and the child must try to identify and type the word that is represented by indecipherable semi-graphics. It would make as much sense to blindfold the kids and say, "okay, go to it, type the word" as to have them try to decode this partial picture.

So skip the Bonus Round and just enjoy the rest of *Kids on Keys*. Because, guaranteed, kids will, along with their parents and teachers!

***—Valerie B. Tamis***

***SPELLBOUND***
***Timeworks***
***C-64***
***Disk***

Weekly spelling tests in elementary school are as unavoidable as peanut butter and jelly sandwiches in the lunchbox. Sometimes they're just as sticky, especially for the student who can't get the "i" before "e" in piece, or put two "c's" in arctic.

Timeworks' *Spellbound* to the rescue! It's a challenging program designed to teach a child basic spelling skills. With amusing graphics, bright colors, and a delightful musical score, *Spellbound* puts the fun back into the fun-damentals of spelling.

Here's how it works. The monitor is randomly sprinkled with letters from one of 20 words your child has entered (perhaps his weekly list?) or from one of the three preprogrammed lists, each progressively more difficult. The word to be spelled is shown at the top of the screen. Using a joystick he guides a Bee around the screen, avoiding incorrect letters in an attempt to "sting" or spell the correct letters of the word in proper order. If he misspells and stings the wrong letter, or crosses back on his own path, the Bee explodes with a crescendo and he must start the game again.

Obviously, the primary object of *Spellbound* is to teach a child to spell words correctly. But man does not live on bread alone! Nor does a video arcade kid of the eighties with a joystick in his back pocket!

So the other point of *Spellbound* is to win the game! And that's where the fun challenge begins. Because with each lesson there are only four bees to zip around the screen. The child must spell each of the words without losing all the bees or else the game is over. As he cruises along on the flight of the bumblebee, he also racks up points because each letter and each completed word are worth varying amounts of points which accumulate for high scores. At any time he can increase the difficulty level, or speed of the bee, from a low and slow 1 to a very rapid 10. And trust me, at level 10 that bee has got to be fleet of wing to beat the clock.

*Spellbound*'s arcade game atmosphere makes it an exciting and fun educational tool, one that just might prompt your Junior or Junior Miss to start spending more time on spelling homework. And if so, who knows where it can end: tomorrow, an A on the weekly test; next month, the National Spelling Bee. Bzzz!

*—Valerie B. Tamis*




***Roam underground passages.***
***READER SERVICE NO. 197***

***DROL***
***Broderbund***
***C-64***
***Disk; joystick***

Having read that *Drol* concerned a witch doctor's curse that threatened to envelop me and my computer, I expected a chaotic mish-mash of a game. Instead I was greeted with an amiable-looking creature armed with smooth round pellets to shoot friendly looking monsters. The object of the game is to rescue a family and their pets, so where's the curse that's supposed to wreak havoc?

In spite of its disillusioning description, *Drol* is an amusing little game. Our hero moves around a four-level underground passageway, shooting and rescuing.

Despite *Drol's* simple outward appearance, it took me over twenty games to get past the first level—and I consider myself an advanced video athlete. Maybe the witch doctor got me, after all.

*—Jamie Marsh*

## *SAT ENGLISH 1*
***MicroLearn***
***C-64***
***Disk***

The most common rite of passage from high school to college is taking the Scholastic Aptitude Test (SAT). Not only is admission to many colleges determined in part by SAT scores, but also awarding of scholarships.

*SAT English I* from Micro Lab contains a variety of the types of questions presented on the SAT. Students can work through any section of the program as either a test or a tutorial. Each work problem or question is presented in multiple choice format.

In the test mode, no feedback is given to the student's answers. At the end of a section, a percentage score is given and all questions answered incorrectly are presented as a tutorial. In the tutorial mode, answers are marked on screen when they are selected. Then the student is told whether an answer is right or wrong and why. Explanations for incorrect answers often give strong clues to correct answers. The student can continue to select answers until he is ready for the next question.

The method of scoring in *SAT English I* is not the same one used for the real SAT. At the end of each section, the program calculates a percentage score based on the number of correct first responses divided by the total number of questions presented (not necessarily answered.) Correctly answering six of twelve questions will earn a score of fifty percent. On the SAT, however, scores are calculated very differently. Students receive four points for each correct answer, no score is given for unanswered questions, and one point is deducted for each incorrect answer. Micro Lab's program places a much higher penalty on guessing and on skipping answers than the actual SAT.

This program drills the student in five areas—sentence completion, analogies, opposites, grammar and usage, and reading comprehension. On the back of the disk are second drills in each area except reading comprehension. A menu allows the student to select any area or to turn the disk over. When a drill is selected from the menu, one-half to a full screen of instructions are given for that particular type of problem. The nine drills contain a total of 148 problems. While a drill, selected from the menu, is loading, the screen shows keys to press to get help, to skip questions, to quit, and to bring back the introductory page. But help is not available on some drills and the introductory page is inaccessible after the first question no matter how many times you press the keys.

*SAT English I* comes with a ten paragraph instruction sheet describing the types of problems and a longer set of instructions for loading and running Micro Learn programs. Nothing is included to describe the SAT or to offer advice on taking it.

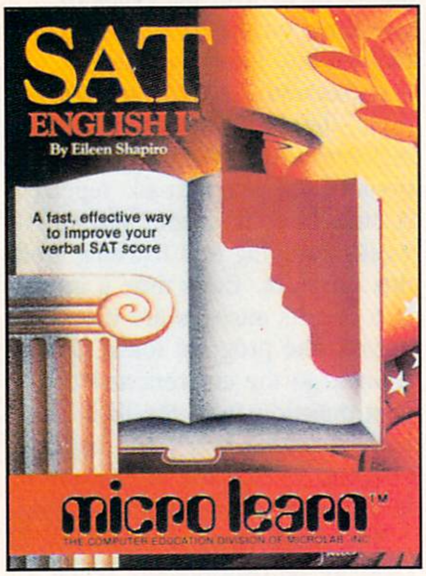
The strengths of this program lie in its ability to give immediate feedback to learners and to acquaint them with the types of questions on the SAT. Although the test and tutorial modes of each section use the same questions, the answers are given in different order when most sections of the program are repeated. A student could spend several weeks studying and learning from this program. Each time a question and its answers are displayed, the program loads its responses to the answers and the next question from the disk. A learner cannot break and list the program to find the answer sitting in some print statement

The weaknesses of *SAT English I* cannot be overlooked. When a question is displayed, ten to thirty seconds will expire as the program loads responses from disk. During this period, no keyboard input is possible. On short questions, you just have to wait, with your answer selected, until the disk stops spinning. Because the program has no built-in timer, the pressure of the SAT is not duplicated. The instructions say speed is not considered since your speed and accuracy increase as you become familiar with the test. At least the test sections should report the time you took along with your score.

Scoring is also a problem. On



*"Awright, who gets the rust resistant beer?"*




***Polish your skills for the big one.***
***READER SERVICE NO. 149***

the SAT, if you can rule out two or three answers as incorrect, there is a benefit (i.e., probable higher score) from guessing. But *SAT English I* penalizes you for guessing unless you narrow the possible correct answers down to two. Then, the chances are, guessing will not hurt your score, but neither will it help.

*SAT English I* can be a valuable study aid when used in conjunction with other SAT preparation tools. I am concerned about its value if used alone because the number of questions is limited and no test-taking strategies specific to the SAT are given. The best value will be obtained from this program if several students can use it as a learning tool.

MicroLearn, 2699 Skokie Valley Road, Highland Park, IL 60035.

*—Richard Herring*

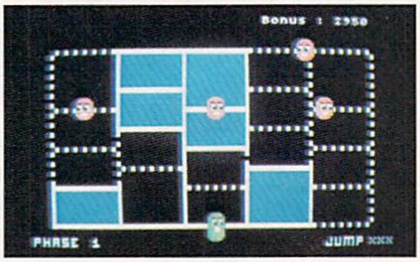
## *TIME RUNNER*
***Funsoft***
***C-64***
***Disk, cassette; joystick, keyboard***

*Time Runner* is somewhat reminiscent of the old arcade flop/home game flop *Amidar;* although *Time Runner* is the superior game, you still might be wise to give it a test run before buying it.

The game field is an overhead view of a maze, but rather than running around inside the maze, you scoot along the top of the maze walls, so to speak. By running the course, you turn dotted lines into solid lines and "box off" rectangular sections. Once you've boxed off the entire screen you go on to the next.

Complicating matters are Defender-Droids, ghostlike foemen who pursue and touch your figure. Once you are touched, you are in "statis" condition and cannot draw further lines. You *can* move on, out of the way of the droids and, once clear, push the fire button to continue the course. Any other time you push the fire button you cause the droids to temporarily flip, which gives you about three seconds of unhindered play.

One of *Time Runner's* greatest strengths is the number of options from which to choose. There are 99 screens that can be accessed at any time, two game speeds, two modes of control, a variable number of jumps and flips and of lives, two scoring modes, and something called "phase time." This means that once you block off half to two-thirds of each section, that space will begin to count down. If the counter reaches zero before you can finish enclosing the space, you will receive no points for the space. In other screens, the droids can undraw what you have drawn.




***READER SERVICE NO. 150***

The graphics are uninspiring, the gameplay familiar. You've been warned. Now let me say that I had a very good time with *Time Runner.*

FunSoft, 28611 Canwood St., Agoura, CA 91301.

*—R.J. Michaels*

## *STARFIRE/FIRE ONE*
***Epyx***
***C-64***
***Disk; joystick***

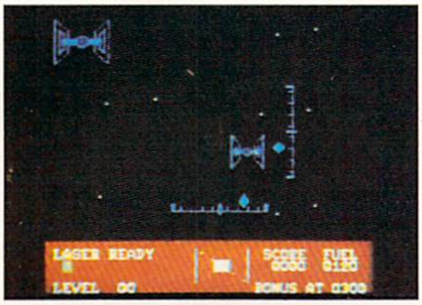
This is another in a series of two-games-in-one from Epyx.

*Starfire* was one of the first arcade games (from Exidy) in which, rather than standing before a screen, you sat in an enclosed booth in order to fry those godless alien scum. No booth here, of course, but the name of the game is the same: a starfield streaks toward you. Among the stars are alien fighters and freighters. Using a radar screen and your joystick controls, you maneuver the enemy into your sights and blast away. Laser fire is limited. Should your guns become overheated, you must merely dodge enemy fire until power is regained.

*Fire One* presents a naval engagement, presumably against godless nazi scum. Each fleet consists of sixteen ships. The top half of the screen displays the sonar view of the entire engagement. The lower half shows the periscope view—a more limited view but a clearer one. With your joystick you control the actions of your sub: firing (your torpedo trajectory depending on whether you're moving or stationary), submerged (you have no periscope view), or surfaced (you are vulnerable).

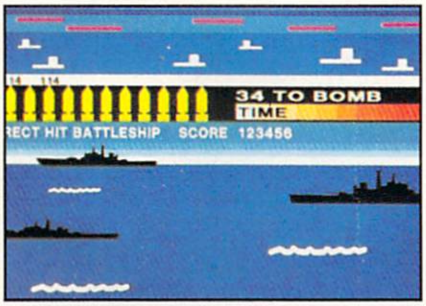
Imagine. Two fleets of sixteen each, dual screens. This game is

as busy and confused as it sounds. Eventually I got the feel of it, but at first I wasn't sure if I was firing at my own ships or the enemy's. But dig those sound effects! Enough bells, explosions,










*Two battles for the price of one from Epyx:* Starfire *(against godless alien scum) and* Fire One *(against a 16-ship enemy fleet).*
*READER SERVICE NO. 151*

sirens, and engine hum to fuel twenty arcades.

*Starfire* is much more sedate and simple. Aim and shoot, aim and shoot. I enjoyed it very much, especially the feeling of motion the onrushing starfield provides.

These are solid programming offerings from Epyx. Nothing new, but twice as much "nothing new" as anyone else is marketing.

Epyx, 1043 Kiel Court, Sunnyvale, CA 94089.

*—Robert J. Sodaro*

## *ATTACK OF THE MUTANT CAMELS*

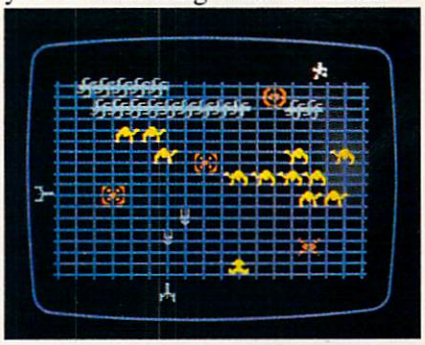
***HesWare***
***C-64***
***Cartridge; joystick***

100 years after the bloody Grid Wars, the deadly droids have returned to menace earth, employing not only their finest weaponry but "bizarre psychological disorientation tactics" as well: "Some pilots have reported sighting what appears to be camels running down the matrix," warns the instruction booklet. Your mission? "...to unleash a stream of white-hot death on the enemy."

Although the gameplay of *Mutant Camels* is similar to that of *Gridrunner* and *Centipede*, designer Jeff Minter has created a game that is much faster, more challenging and more demanding of strategy than either of those. There is a fiendish mind at work here.

The screen is a grid. The player controls a ship with full horizontal movement and three quarters vertical movement. Firing upwards, the player must destroy a descending file of cameloids and droids while avoiding their dropped bombs. The player must also watch for the destructive Y Beam, which a Zapper controls while roaming the lower portion of the screen. A vertically roaming Snitch (a traitorous humanoid) will point you out to the Zapper. After wave 6, droids may attack diagonally. Further complicating matters are Deflexors, which will cause your salvos to ricochet. There are 20 levels in all; the player can choose any of the first six. After that, he or she must earn the right to advance.

Obviously, there is a lot to keep track of here. Action is very fast, requiring a high degree of skill. Sound effects are, how shall I say, cosmic. The graphics, with so much going on, are attractive but little more than serviceable, so if you're wondering what a mutant camel really looks like, forget it. Despite, this, *Mutant Camels* is, all in all, a fine game.



*READER SERVICE NO. 152*

HesWare, 71 Park Lane, Brisbane, CA 94005.

*—Scott Kincaid*

## *POGO JOE*

***Screenplay***
***C-64***
***Disk, cassette; joystick***

If nothing else, *Pogo Joe* will make you smile. The game has personality to spare: toe-tappin' music, hop-toad gameplay, and a goofy looking kid with a village idiot's grin, clutching his pogo stick for dear life.

Joe's object is to turn various cylinder tops from one color to another with his pogo stick while




*READER SERVICE NO. 146*

avoiding hopping creatures. Sound like *Q*Bert?* You bet. Though *Pogo Joe* has none of *Q*Bert's* bizarre and splendid sound effects, it has enough gameplay variations to make it worth your while, if this type of game is to your taste.

For example, Joe can catch some of the creatures for points, others he must avoid. Some creatures will hop on cylinders that Joe has already changed and change them back. When Joe jumps on the occasional flash square, all creatures are destroyed. With the fire button, Joe can make double jumps, which are often necessary since cylinder clusters are far apart. Game structure variations include the number of players (one or two), initial screen (there are 64 in all), Joe's speed and the creature's speed (in increments from one to six), and the choice of whether or not the creatures have the ability to double jump. There is also a pause feature.

Unlike that other game, Joe cannot fall to his doom outside the playfield. Also, the basic configuration of the cylinders changes with each screen. Graphics are excellent, but control was a serious problem: sluggish, confusing, haphazard. I never really got accustomed to them. That's a serious reservation, but if it can be overlooked, *Pogo Joe* is a fine game. The multiple speed variations make it perfect for a home with lots of kids.

Screenplay, Box 3558, Chapel Hill, NC 27514.

*—Tim Moriarty*

## *ALL ABOUT CHANUKAH*
***Davka Corporation***
***C-64***
***Disk***

This package, aimed primarily at the Jewish segment of our readership, presents a concise summary of the history and traditions associated with the celebration of Chanukah. Targeted for the mid-elementary school age group, it will also be of interest to younger children as well as adults, depending on their past contact with their heritage. The program is designed to act as an introduction for those with a minimum of previous contact with their cultural heritage and as a review for younger attendees of Hebrew day schools. Non-affiliates with a general interest in the subject matter will find enlightenment as well.

The program consists of eight (a key number for this celebration) parts which are menu-driven. The single-page instruction sheet was more than adequate, as the program practically runs itself. Three parts present the historical and traditional background of the celebration. A fourth part, using a combination of text and graphics, presents the lighting of the Chanukah Menorah or candelabra. Another section presents the traditional blessings which are recited during the Menorah-lighting ceremony. There is even a twenty-question, multiple-choice quiz to reinforce the more significant facts. The program provides a built-in diversion, a traditional dreidel (spinning top) game. Finally, for the philosophical, a brief compilation of Rabbinic wisdom on the holiday of Chanukah based on Talmudic and Midrashic sources.

The overall performance of the package was good, although it made minimal use of the Commodore 64 sound and graphics capabilities. The rendition of the traditional melodies was done using only one of the three voices available. The Chanukah Menorah was nicely rendered using character and sprite graphics. The presentation of the traditional candlelighting blessings could have used a bouncing ball on the English transliteration for those who cannot read Hebrew.

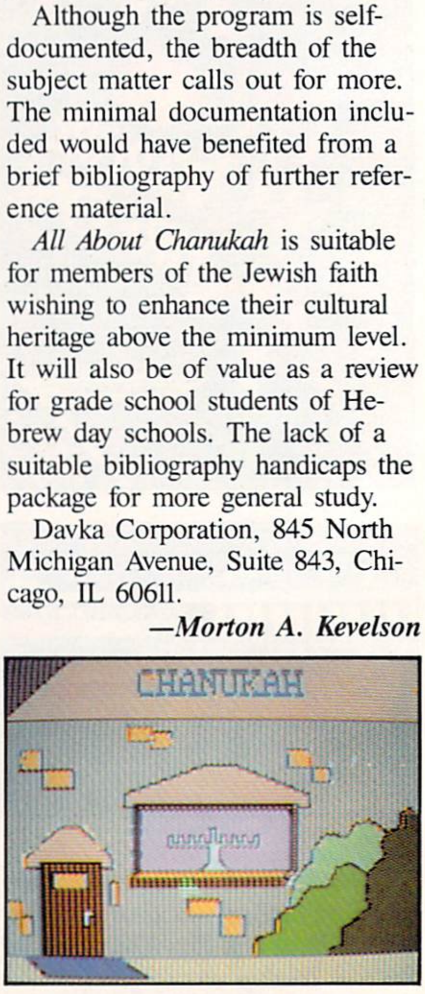
The Dreidel game, although lacking the flash and pizzazz of some of the arcade spectaculars, was quite effective. It is particularly appealing to the younger members of the family.

Although the program is self-documented, the breadth of the subject matter calls out for more. The minimal documentation included would have benefited from a brief bibliography of further reference material.

*All About Chanukah* is suitable for members of the Jewish faith wishing to enhance their cultural heritage above the minimum level. It will also be of value as a review for grade school students of Hebrew day schools. The lack of a suitable bibliography handicaps the package for more general study.

Davka Corporation, 845 North Michigan Avenue, Suite 843, Chicago, IL 60611.

*—Morton A. Kevelson*




*READER SERVICE NO. 147*




# THAT'S MY PROGRAM !?!?!

## THE MYSTERIOUS UNDERWORLD OF BASIC PROGRAM STORAGE

*By Michael Kleinert and David Barron*

Most of us enter a program, type the word RUN, and don't care about anything except whether or not the program works. What the computer has done to each line you type in is just about as important to you as the time of night (morning?) you finished it. In this article we will summon the raw strength and courage necessary to venture into the mysterious underworld of BASIC program storage (get psyched!). Here we will encounter beings such as links, pointers, tokens, and other beasties.

The Commodore 64 stores your BASIC program starting at memory location 2048 (this is $0800 in hexadecimal) and ranging up to 40959 ($9FFF hex). The end of BASIC may be reduced by the presence of certain ROM cartridges which add functions to the computer. Without any cartridges present, you have approximately 39K available to store your programs; however, every variable used by a BASIC program takes up part of this memory. When programming, you must keep this in mind, for the more variables a program uses, the less space is available for program lines.

Many of you might think that the computer stores a program in memory exactly as you type it on the screen. This is not so. Although this is one possible method, a more efficient one was chosen. Since a program is made up entirely of the BASIC language's limited vocabulary, the computer has this vocabulary stored in its memory and each word (PRINT, GOTO, FOR, etc.) is assigned a value. This value is referred to as a token. This enables the computer to represent each RESERVED WORD with a single byte in memory. Only BASIC reserved words are stored in this fashion; things such as numerical values, characters in quotes, and variable names are stored character for character. Unrecognized groups of characters are stored as their individual character codes in memory, each taking up one byte. A list of the character codes and tokens which are used for program storage is provided in Table 1. Notice that tokens for BASIC reserved words are all greater than 127.

The computer uses an additional four bytes of memory before the contents of each line of your program. The first two bytes determine a LINK which "points" to the byte immediately following the line (the LINK of the next line). This is a memory address stored in a binary format. The first byte is referred to as the LO byte of the address, and the second as the HI byte. To calculate the address, simply use the following formula:

```
ADDRESS = LO + (HI * 256)
```

Preceding the final link is a zero byte. This is at memory location 2048, so the link of the first program line starts at 2049. The computer uses links to rapidly search through a program for a particular line. Without them, any operations involving line numbers such as GOTO, GOSUB, and BASIC editing would be much slower.

The next two bytes hold the line number in the same LO byte, HI byte format.

```
LINE = LO + (HI * 256)
```

After the line number is the tokenized text of the line followed by a zero value which indicates the end of the line. The byte following the zero is the link for the next line. This pattern continues until the end of your program. Here the zero is followed by two additional zeros in place of the usual link.

Let's take a look at how the following short program is stored in memory in a Commodore 64.

```
10 PRINT"DEMO"
20 FORLOOP=1TO999:NEXTLOOP
30 END
```

On page 92 we show you how it would appear if it were typed in with no spaces.

Everything we have discussed is also true for the VIC 20, except for the memory locations which hold BASIC programs. On a VIC with no added memory, BASIC starts at 4096 ($1000 hex). You can find the start of BASIC on any VIC or 64 by typing: PRINT PEEK (43) + (PEEK(44) * 256). This will really return a value one greater than what we have discussed because it actually points to the link for the first line.

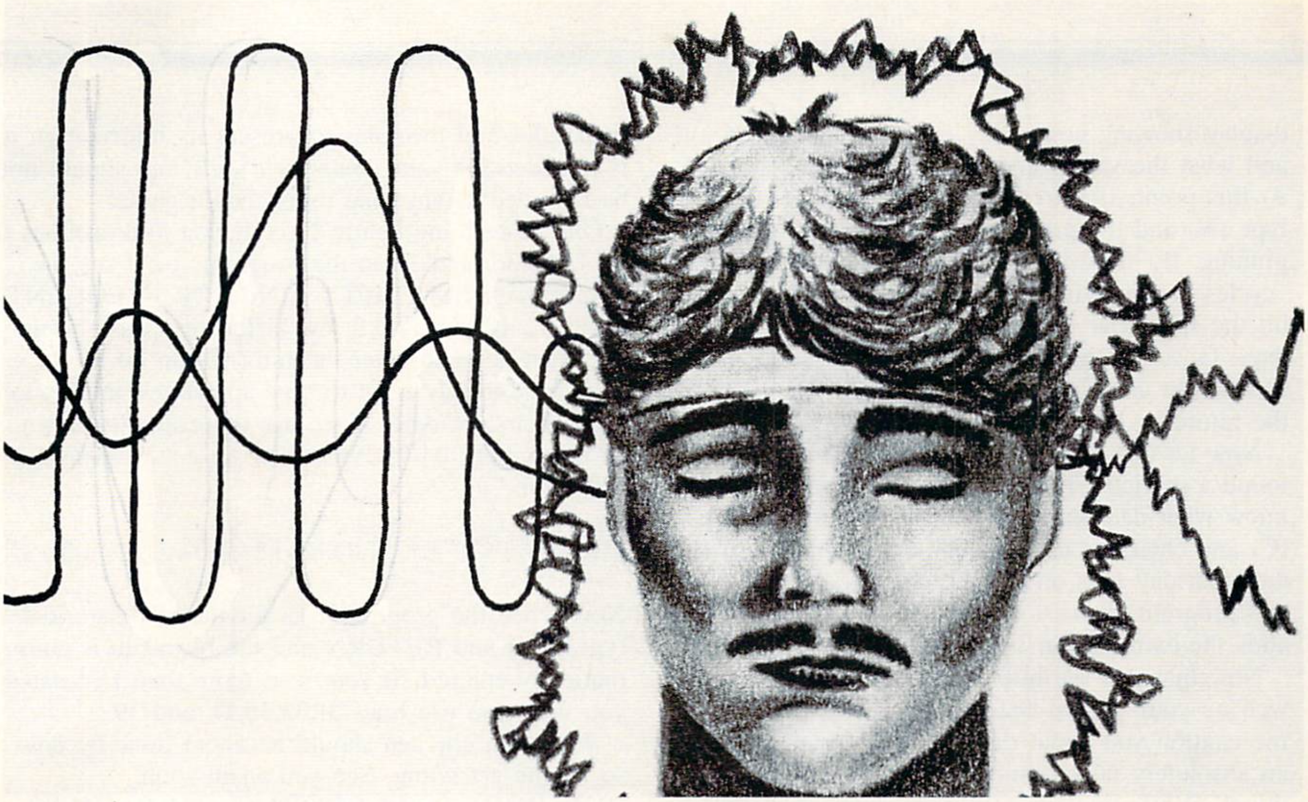
You may have realized several things from what we have explained. Spaces between statements in a line occupy one byte each, but one SPACE is automatically placed after each line number when a program is LISTed and does not take up any extra memory. Every line requires two bytes for the link, two bytes for the line number, and one for the zero byte—a total of five bytes.

Therefore, when you use a colon (:) to put an additional statement onto an existing line, instead of adding a new line, one byte will be used for the colon and you will save four bytes. In a larger pro-

*Continued on page 92*

# TABLE 1

| CODE | CHARACTER/ KEYWORD | CODE | CHARACTER/ KEYWORD | CODE | CHARACTER/ KEYWORD | CODE | CHARACTER/ KEYWORD |
|---|---|---|---|---|---|---|---|
| 0 | End of line | 92 | │ | 150 | DEF | 177 | > |
| 1-31 | Unused | 93 | ] | 151 | POKE | 178 | = |
| 32 | Blank Space | 94 | ↑ | 152 | PRINT# | 179 | < |
| 33 | ! | 95 | ← | 153 | PRINT | 180 | SGN |
| 34 | " | 96-127 | Unused | 154 | CONT | 181 | INT |
| 35 | # | 128 | END | 155 | LIST | 182 | ABS |
| 36 | $ | 129 | FOR | 156 | CLR | 183 | USR |
| 37 | % | 130 | NEXT | 157 | CMD | 184 | FRE |
| 38 | & | 131 | DATA | 158 | SYS | 185 | POS |
| 39 | ' | 132 | INPUT # | 159 | OPEN | 186 | SQR |
| 40 | ( | 133 | INPUT | 160 | CLOSE | 187 | RND |
| 41 | ) | 134 | DIM | 161 | GET | 188 | LOG |
| 42 | * | 135 | READ | 162 | NEW | 189 | EXP |
| 43 | + | 136 | LET | 163 | TAB( | 190 | COS |
| 44 | , | 137 | GOTO | 164 | TO | 191 | SIN |
| 45 | – | 138 | RUN | 165 | FN | 192 | TAN |
| 46 | . | 139 | IF | 166 | SPC( | 193 | ATN |
| 47 | / | 140 | RESTORE | 167 | THEN | 194 | PEEK |
| 48-57 | 0 to 9 | 141 | GOSUB | 168 | NOT | 195 | LEN |
| 58 | : | 142 | RETURN | 169 | STEP | 196 | STR$ |
| 59 | ; | 143 | REM | 170 | + | 197 | VAL |
| 60 | < | 144 | STOP | 171 | – | 198 | ASC |
| 61 | = | 145 | ON | 172 | * | 199 | CHR$ |
| 62 | > | 146 | WAIT | 173 | / | 200 | LEFT$ |
| 63 | ? | 147 | LOAD | 174 | ↑ | 201 | RIGHT$ |
| 64 | @ | 148 | SAVE | 175 | AND | 202 | MID$ |
| 65-90 | A to Z | 149 | VERIFY | 176 | OR | 203-254 | Unused |
| 91 | [ | | | | | 255 | π |




# I GOT BIORHYTHM

***By B.W. Behling***

The theory of Biorhythm states that humans are influenced by three "cycles" that start at birth:

The Physical cycle: 23 days
The Emotional cycle: 28 days
The Intellectual cycle: 33 days

These "cycles" can be positive, negative, or neutral in their effect on how we react to our surroundings on any given day. Represented mathematically, these "cycles" range from +1 to −1, +1 being a "good" or "up" influence and −1 being a "bad" or "down" influence. A day when a "cycle" is at or close to "0" is a "critical" day since that "cycle" is undergoing a change and can cause wide swings in our behavior.

Now, whether you believe in biorhythms or not, it is an interesting theory that adapts itself well to the home computer. If you're a believer or if you just want an interesting program to run at your next party, take the time to type this in and experiment with it. Who knows? I may have something here.

Before I begin with an explanation of how the program works, let's get it typed. While you're busy with that I'm going to go grab a glass of iced tea and see what my wife has planned for supper. I'll meet you back here in 30 minutes.

Got it typed in yet? Good! Looks like I'm having chicken tonight. Anyway, before you run the program take a minute and save it just in case it goes haywire. After all, nobody's perfect.

O.K., now run it. Pretty neat opening graphics, huh? Now enter your birthdate and hit return. Be sure to follow the example shown on the screen for all dates. If you try to enter a date like April 9, 1963 using the format 4/9/63 or 4/9/1963, you'll get an error message later on and have to reenter all your dates. Next enter the date you want the biorhythm for and hit return. Now you should have a

display showing how many days you have been alive and what the values are for each of your "cycles." At this point, if you want to check another date just type (N) and the program will go back to the beginning. If, however, you are curious about how the "cycles" are moving in relation to each other, just hit the space bar to get a forecast of the next seven days. Once you have the seven-day forecast on the screen you can continue to take seven-day peeks at the future by hitting the space bar.

Now let's say you've gone ahead a few weeks and found a particularly good or bad day and want to know what date that day will fall on. Simply type (C) and enter the day number, hit return, and the date that day falls on will be displayed. By the way, this program allows for leap years and automatically adds the extra day in February.

Now that you see how to use the program and are well on your way to becoming a biorhythm buff, let me caution you about one thing. Biorhythms are *not* an absolutely failsafe method of predicting anything!! While this program uses common mathematical rules and formulas to present its information it is intended for your *amusement only* and should not be regarded as anything more than a game!

One more thing before I go. If you look at lines 30,32,34,36, and 38 in the program you'll see the statement "REM EMPTY LINE FOR FREQUENT DATES". If you have a few birthdates you'll be using often you can enter the information on these lines so you only need to type someone's initials to enter their birthdate. If we use the example shown on the screen, 04/09/1963, then we could change line 30 to:

```
30 IFBI$="EX"THENBI$="04/09/1963"
```

Now when the program asks for the birthdate just type (EX) and RETURN and the birthdate is automatically entered. If you need more than five dates you may also use lines 31,33,35,37, and 39.

Well, that chicken should be about done by now so let me get going. See you again soon!

***SEE PROGRAM LISTING ON PAGE 80***




ARTIST: JOHN COSTANZA

# Math Master

## for the C-64

***By Bob Lloret***

*Math Master* is a program designed to help you practice and improve your math skills. You can drill yourself on addition, subtraction, multiplication, and division on three different levels. For our younger readers, I suggest starting at the lower level and working on up.

After typing in the program, save a copy to disk or tape in case of any typing errors. Type "RUN" and the program should start immediately.

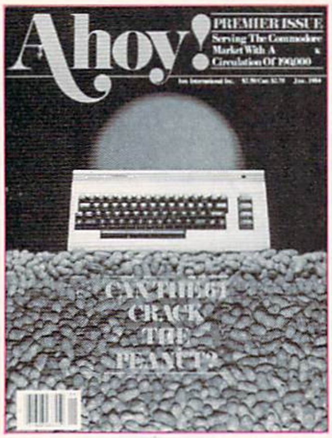
The screen will ask for the player's name. Type in your first name and hit return. The rest of the screen will display the main menu. Select a number from 1 to 4, then hit return. The next screen will display a submenu from which you must select a skill level. Press 1 to 3 and hit return.

Depending on the number selected, the program will then take you through a series of 10 problems. The top of the screen will display which type of

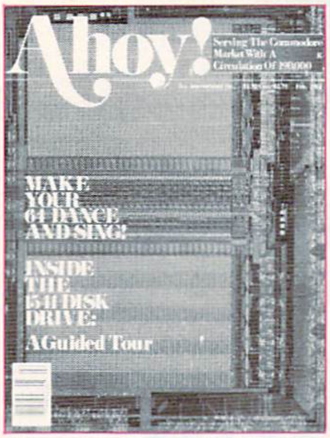
*Continued on page 71*

# Ahoy!

## Back Issues




**ISSUE # 1—JAN. '84 $4.00**
The 64 v. the Peanut! The computer as communications device! Protecto's Bill Badger interviewed! And ready to enter: the Multi Draw 64 graphics system! The Interrupt Music Maker/Editor! A Peek at Memory! Programming Sequential Files on the C-64!



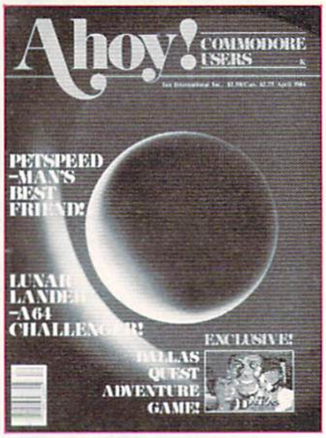

**ISSUE # 2—FEB. '84 $4.00**
Illustrated tour of the 1541 disk drive! Synapse's Ihor Wolosenko interviewed! Users groups! Artificial intelligence! And ready to enter: Music Maker Part II! Night Attack! Programming Relative Files! Screen Manipulation on the Commodore 64!

**Sure, you're a dedicated *Ahoy!* reader now—but what about any issues that might have seen print *before* you climbed on board? Can you compute on in total confidence without a complete collection of *Ahoy!* and the programming strategies and software and hardware analyses each issue provides? Can you rest secure in your knowledge of the Commodorean world without reading all our feature articles, industry updates, and regular columns? Can you hold onto your four bucks in good conscience knowing how short we are on closet space?**

## Order your back issues TODAY!




**ISSUE # 3—MAR. '84 $4.00**
Anatomy of the 64! Printer Interfacing for the 64 and VIC! Educational software: first of a series! Commodares! And ready to enter: Space Lanes! Random Files on the 64! Easy Access Address Book! Dynamic Power for your 64!




**ISSUE # 4—APR. '84 $4.00**
Petspeed and Easy Script tutorials! Printer interfacing and educational software guide continued! Lower case descenders on your 1525! Laserdisc! The Dallas Quest Adventure Game! And ready to enter: Apple Pie! Lunar Lander! Name that Star!

**Send coupon or facsimile to:**
*Ahoy!* Back Issues, Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001

Ahoy!

Please Send Me The Following:
_____ Copies of issue number _____
_____ Copies of issue number _____
_____ Copies of issue number _____
Enclosed Please Find My Check or Money Order for $__________
(Outside the USA please add $1.00 for every copy)

NAME__________
ADDRESS__________
__________
CITY__________
STATE__________
ZIP CODE__________

# RUPERT REPORT




## An Article on Arrays and Subscripted Variables

### *By Dale Rupert*

This month we will examine the most sophisticated variable type in BASIC. We will investigate subscripted variables and arrays.

Until now we have dealt with simple variable types: integers, floating point numbers, and strings. Let's look at an example involving the stock market to see where a more powerful variable type is useful.

If you want to enter and graph the weekly average of one stock over a month's time, the following program would work:

```
10 INPUT "WEEK 1 AVERAGE";W1
20 INPUT "WEEK 2 AVERAGE";W2
30 INPUT "WEEK 3 AVERAGE";W3
40 INPUT "WEEK 4 AVERAGE";W4
50 PRINT:PRINT "WEEK 1";
60 FOR S=1 TO W1:PRINT"*";:NEXT
70 PRINT:PRINT "WEEK 2";
80 FOR S=1 TO W2:PRINT"*";:NEXT
90 PRINT:PRINT "WEEK 3";
100 FOR S=1 TO W3:PRINT"*";:NEXT
110 PRINT:PRINT "WEEK 4";
120 FOR S=1 TO W4:PRINT"*";:NEXT
```

Obviously, this program has some shortcomings. For example, if we expanded it to cover a year's worth of averages, the program would be twelve times as long. Using the powerful screen editor of the Commodore, that wouldn't require a great amount of additional typing, but what do we use for variables after W9?

The Commodore can't distinguish between W1 and W10. (To convince yourself, type W1 = 5: PRINT W10.) Consequently we would have to use other letters. There are probably enough combinations. The first character must be a letter (26 possibilities). The second character is either a letter, a digit, or nothing (37 possibilities). That totals 962 choices, almost. The Status Function ST, as well as TO, ON, IF, OR, and TI are not valid variable names. (Try typing ST = 5). Do you know of any others?

So it would be possible to come up with about 80 years' worth of variable names, but there must be a better way. And of course, there is: subscripted variables.

With simple variables, we have to specify each item by name. With subscripted variables, we may identify items by number. That is a big advantage since the numbers themselves may be identified by a variable.

Let's see what that preceding bit of confusion really means. Our stock market example above could be written with subscripted variables as follows:

```
10 FOR N=1 TO 4
20 PRINT "AVERAGE OF WEEK #" ; N
30 INPUT W(N)
40 NEXT N
50 FOR N=1 TO 4:PRINT "WEEK" ; N
;
60 FOR S=1 TO W(N) : PRINT "*";:
NEXT S
70 PRINT : NEXT N
```

Lines 10 through 40 perform the same function here as in the previous program. Now we are entering weekly values into the variable W(N) in line 30. Actually W(N)—pronounced "W sub N"— represents an array of variables: W(1), W(2), W(3), and W(4) in this this program. Each of these variables stores an independent, floating-point value. We may select any one of them by specifying its subscript. For example, we could add these lines to the program above:

```
80 INPUT "WHAT WEEK'S AVERAGE DO
YOU WANT" ; N
90 PRINT "THE AVERAGE OF WEEK #";
 N; "IS"; W(N)
100 GOTO 80
```

Now we are able to easily recall any value we have previously entered. Adding this feature to the first program without subscripted variables would be very cumbersome indeed. Try it if you don't believe it.

When the computer sees PRINT W(N), it first finds the current value of N. It then knows which element of the array W to print. Think of W as the surname of a set of memory cells. W(1) is the name of one specific cell, W(2) is a different cell, etc.

The *Commodore 64 User's Guide* gives a good presentation of subscripted variables beginning on page 95. You may think of W(1) the same way you think of the variable W1. The main difference between the subscripted variable and the simple variable is in the flexibility with which we may identify each one. As we have seen, it is much easier to specify a particular item out of a group of many such items with subscripted variables than it is with simple variables.

When is it desirable to use subscripted variables instead of simple variables? Whenever there are several items that are similar in nature, on which similar types of operations are to be performed, subscripted variables are generally preferable.

It is now easy to expand our stock program to handle as many weeks as desired. Simply change the number 4 in lines 10 and 50 to whatever value is desired. You will find that if you use a number larger than 10, the computer responds with "? BAD SUBSCRIPT ERROR IN 30." The problem is that the computer only allows room for subscripted variables with subscripts of 0 through 10.

The solution is that we may make room for even larger arrays by using the DIM statement at the start of our program. If we add

```
5 DIM W(52)
```

to our program, BASIC sets aside enough memory for us to store up to 53 values—W(0) through W(52). The DIM statement does not assign values to this array (although the values are initialized to 0), but allocates the storage space for future use.

The program on page 99 of the *Commodore 64 User's Guide* simulates the rolling of dice. Six elements of the array F store the total number of times that a particular face value is "rolled." Study it to see another way that arrays may be used.

Listing 1 presents a program in a similar vein. It simulates the random falling of a ball Pachinko-style through a maze of nails.

The ball starts at the top center of the screen. It hits a nail and falls either left or right. It hits another nail and falls to the left or right again. This continues until it reaches the bottom row of nails, at which point it falls straight down into a trough.

The troughs near the center are more likely to be filled first. It is possible, but less probable, for the ball to always fall to the right and end up in the rightmost trough. The result is a (somewhat) bell-shaped graph of the number of balls in each location.

The array N stores the number of balls in each trough. N(0) ("N sub zero") is the number in trough 0 on the left, and N(39) is the number of balls in trough 39 on the right. The ball starts at row 0 and column 20. Line 120 picks a random number 0 or 1. The variable DIR is given the value −1 or 1 depending on the random number. Line 140 increments (or decrements) the variable COL to simulate the ball's motion to the right (or left.)

This program uses the cursor-positioning routine described in Rupert Report #1 (in the January *Ahoy!*). The subroutine at line 2000 puts the machine language code into memory. The subroutine at line 1000 is called when new values of ROW and COL have been determined in line 140. Once the cursor is positioned, line 150 prints the ball. Line 180 erases the ball and replaces it with a dot unless the ball is below the stack of balls already in that column.

The ROW is incremented in line 190 and the process is repeated. Once the ball reaches ROW 20, its final column is stored in COL. Line 200 adds one to the subscripted variable N(COL) to indicate that one more ball has fallen into that trough.

For example, if COL equals 4 when the ball reaches ROW 20, N(4) is increased by one. If the value in N(4) was two, it is now three, indicating that three balls have fallen into the fourth trough.

Line 210 does the mathematical job of determining how high the stack of balls is on the screen. There are 25 rows on the screen, 0 at the top through 24 at the bottom. If N(4) is now three, a new ball should be placed in row 22: ball #1 in row 24, #2

in row 23, and #3 in row 22. Consequently 25 — N(COL) gives the correct value of the new row.

The subroutine at line 1000 positions the cursor, and line 220 prints the ball on the stack. Line 230 calls the sound generating routine at line 3100. Finally, line 240 takes us back to line 100 where a new ball is dropped.

Just think of N(0) thru N(39) the same way you think of the simple variables N0 or A5 and this program should be no more difficult to understand than if only simple variables were used.

Just as simple variables may be floating-point, integer, or string types, subscripted variable types may also be specified. An array of string variables might be represented by N$(I). This simple program shows how alphanumeric information is stored and used in subscripted variables:

```
5 N=4
10 FOR I=1 TO N:READ N$(I),A(I):N
EXT I
20 DATA AMY,25,ABE,12,ARTHUR,30,A
NNE,5
30 REM - SORT BY NAME -
40 FOR I=1 TO N
50 FOR J=I TO N
60 IF N$(I)>N$(J) THEN GOSUB 1000
70 NEXT J
80 NEXT I
90 FOR I=1 TO N:PRINT N$(I),A(I):
NEXT
100 END
990 REM - SWAP NAMES AND AGES -
1000 T$=N$(I):N$(I)=N$(J):N$(J)=T
$
1010 T=A(I):A(I)=A(J):A(J)=T
1020 RETURN
```

Line 5 indicates that there are four sets in the DATA

list. Line 10 reads a name and an age from line 20. The first name is stored in N$(1) and the corresponding age is stored in A(1). This is repeated similarly for N$(2) and A(2) through N$(4) and A(4).

Lines 40 thru 80 perform a bubble sort on the names. If a name is greater alphabetically than another name further in the list, the two names and corresponding ages are swapped by the subroutine at line 1000. After line 80 is completed, the names are in alphabetical order. Lines 90 prints the sorted names and ages.

This is not the most efficient method of sorting, but that is the topic of another article. This is an example of a program that would be absurdly tedious to write without subscripted variables.

Listing 2 shows how to set up and shuffle a deck of cards. Once again, this is a prime example of the type of program where subscripted variables are essential.

Memory space for three arrays is allocated by line 20. CARD is a numeric array in which each element stores a number from 1 to 52, with no two numbers the same. S$ is a string array which is assigned the graphics characters for diamonds, clubs, hearts, and spades in lines 100 and 110. V$ is given the string values corresponding to the face values of the cards in lines 120 and 130.

Two functions are defined in lines 30 and 40. FNSUIT(N) converts a card number from 1 to 52 into a suit number. Cards 1 through 13 represent the ace through the king of diamonds. Similarly cards 14 through 26 are clubs, cards 27 through 39 are hearts, and 40 through 52 are spades. The suit numbers are 0 through 3 for diamonds, clubs, hearts, and spades respectively.

FNVLU(N) converts a card number from 1 to 52 into its face value. The face values range from ace, which is one, to king, which is 13.

Line 50 calls the subroutine at line 100 which defines the elements of the arrays S$ and V$. Line 60 initializes the CARD array. It assigns a value of 1 to CARD(1), 2 to CARD(2), up through 52 to CARD(52). Think of CARD(1) as the first card on the deck and CARD(52) as the last card. The subscript will always indicate the position of the card in the deck. The value of the subscripted variable always tells which card is at that position. For example, after the deck is shuffled, the third card down might be the ace of clubs. The card value of the ace of clubs is 14. Therefore CARD(3) equals 14. Also CARD(52) equals 5 when the bottom card on the deck is the 5 of diamonds.

Line 70 calls the subroutine at line 200 to show the cards in order from top to bottom. In line 210, FNVLU(1) is 1 and FNSUIT(1) is 0 since card number 1 is the ace of diamonds. Thus line 210 first prints V$(1) and S$(0) which are "A" and the diamond symbol respectively. This is repeated for all 52 cards.

Line 80 calls the shuffle routine at line 300. This routine takes each card in turn and exchanges it with a randomly chosen card in the deck. Line 310 picks a random card position from 1 to 52. That card's value is swapped with the card at position N in line 320. After all cards have been exchanged, the program goes to line 90, where the newly shuffled cards are displayed.

These subroutines may be easily incorporated into your own programs. Do you want to know what the fifth card from the top of the deck is? Simply type GOTO 500 and respond with 5.

If you've always wanted to play with a stacked deck, here is your chance. How many hands must your computer deal before it comes up with four aces? Are you able to modify this program to include a couple of jokers?

For those of you for whom this discussion has been review, and for anyone else who is interested in the innards of the computer, let's see how arrays are stored in memory. According to the *Commodore 64 Programmer's Reference Guide,* there is a pointer called ARYTAB at decimal locations 47 and 48 showing the start of array storage in BASIC. The pointer at 49 and 50 called STREND shows a value one greater than the end of BASIC array storage.

*Continued on page 94*



*"I think this artificial intelligence stuff is a bunch of nonsense. Don't you, Smedlow?"*

# AIR ASSAULT

*Continued from page 35*

higher the amount of points you'll receive for each one.

When the bombs start falling, line up your gun sight with the bomb and press the fire button on the stick. If your reflexes are good you'll explode the bomb in midair and save your city. If not, you'll miss and the bomb will fall, wiping out a few houses. If you miss more than eight bombs your city will be considered wiped out. The program will continue until 40 bombs have been dropped. After 40 the screen changes to give you the final totals. You will then automatically go to the next round.

For those interested, the following shows the program flow.

| | |
|---|---|
| Lines 100-130 | Title & author information. |
| Lines 160-225 | Title screen. |
| Lines 230-270 | Redefines characters, creates sprites, draws main screen. |
| Lines 280-410 | Main loop—Moves bombs, reads joysticks and detects collisions. |
| Lines 420-477 | Subroutine to shoot down bombs. |
| Lines 480-570 | Subroutine to destroy bombed city. |
| Lines 1200-1330 | Subroutine to give final results of assault. |

*SEE PROGRAM LISTING ON PAGE 82*

# MATH MASTER

*Continued from page 65*

math you selected, along with the level of play and the problem number. Under this information a black rectangle is drawn. There's a short pause, and the current problem is displayed in the rectangle. At the same time, the built-in Commodore time clock is displayed on the bottom of the screen. At level 1 you will have 15 seconds to give your answer, at level 2 you have 12 seconds, and at level 3 you have 10 seconds. If you type in the correct answer before the timer runs out, a message will appear just below the problem telling you it's correct and indicating your total number of correct answers. If you type in the wrong answer or the timer runs out, a message will appear telling you it's incorrect and the correct answer will be displayed. The program will then go on to the next problem and continue until you've completed 10 problems.

Upon completion of the 10 problems, the program gives your results and makes a comment according to the number of problems answered correctly. You have the option of ending the program or trying again. If you try again you have the same options to choose from.

For future programmers, the program flow is as follows:

| | |
|---|---|
| Lines 10-20 | Title & author information. |
| Lines 90-220 | Title screen and type of math choice. |
| Lines 230-330 | Difficulty level. |
| Lines 360 | Conditional branch (type of math chosen). |
| Lines 500-610 | Creates addition problems. |
| Lines 620-665 | Built-in timer, records answers. |
| Lines 670-750 | Correct answer response. |
| Lines 760-830 | Wrong answer response. |
| Lines 840-900 | Expired time response. |
| Line 1000 | Starts subtraction problems. |
| Line 1500 | Starts multiplication problems. |
| Line 2000 | Starts division problems. |
| Line 3000-3250 | Gives final results of problems and remarks on your results. |

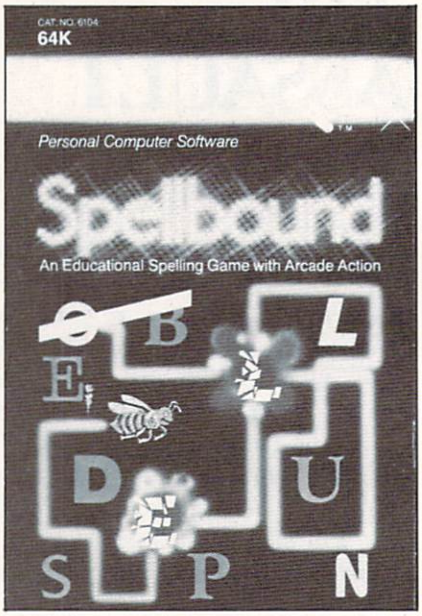
*SEE PROGRAM LISTING ON PAGE 85*

# EDUCATION

*Continued from page 43*

In *Letter Games*, a program for elementary school age children, input is limited to a few specific keys. Every key that is not needed to give answers is disabled. *Letter Games* includes three matching games for the child to play. In each one, the child must pick which one of four letters is different. Next to each letter is a highlighted number from one to four. Only those keys, one to four, are recognized by the program. Once the child understands this system, he does not have to search the keyboard for the right answer.

There are some high quality programs which, unfortunately, do not follow this advice. Spinnaker's *Kinder-Comp* is a group of six programs for pre-kindergarten to elementary school age children. In the program named *Sequence*, the next number in a sequence like "6 7 8" must be typed. The program will accept almost any input including letters and punctuation marks.




***In Timeworks'* Spellbound, *the child joystick-guides a bee to sting the scattered letters of a word in the proper order. As his skills develop, new words can be added. (See review elsewhere in this issue.)* READER SERVICE NO. 186**

A child might become confused when he types the word "nine" and his answer is rejected.

The age group at which educational programs are aimed should establish how they deal with input. For pre-kindergarten children, a single keystroke should be all that is necessary. Even having to hit the return after pressing a letter may be too much. To aid these young children's explorations, some programs allow every key to have a purpose. Software for kindergarten age and slightly older children may require the use of only numbers or letters. For still older kids, who will type in whole lines, the program must allow editing. Remember, we are not all touch typists. The child will have to look up from the keyboard to see his typing errors. If he is forced to press return because he cannot backspace to correct an error, he will feel, and I agree, that the program is unfair. Many programs, like *KinderComp,* allow your child to correct any eror he sees before he hits return.

Are parents or children allowed to modify the program? Even if you do not have the skills, you may be surprised at what your child can do. And you can bet that if the program accepts supplemental data files, somebody will write them. You may even find enhancements to the program right here in *Ahoy!* An educational software budget really can be stretched by programs which grow with the child. The simplest example is the hangman-type program which allows you to put in your child's spelling list each week. It may have value for several years.

Two good examples of programs you can modify are Timeworks' *Spellbound* and Lightening's *MasterType.* In *Spellbound,* letters are scattered over the screen. Using a joystick, the child guides a bee to sting the letters of a word in the right order. As the child's skills develop, new words can be added to the program. In *MasterType,* you defend a spaceship in the center of the screen from letters and words which attack from the corners. The program comes with lessons similar to a typing course, leading you through home, third, and bottom row exercises before throwing full words at you. It also allows you to design your own set of words. This could be a list of assembly language op-codes, the streets in your town, names of friends, or a list of vocabulary words. In this way, you can use *MasterType* to learn more than just typing skills.

For some educational software, supplemental programs are available. The *SAT Preparational Series* from Krell helps high school students study for their college boards. Topics covered include vocabulary, word relationships, sentence completion, reading comprehension, math, and tests of standard written English. For the math and vocabulary sections, the program will generate learning coefficients based on scores previously saved on the disk. When the program is run again, those areas in which the child performed poorly will be repeated and those in which she did well will be covered less often. Available options for this program includes a *Diagnosis and Study Prescription* to keep track of performance by skill type, a *Worksheet Generator* to create worksheets on your printer, and a *Vocabulary Builder.*

Next, think about who will use the program. Adults have read about the delicacy of data tapes and disks. But your child may not understand that the diskette really should not be bent, or his small hands may not manipulate the disk drive easily. Think about how many electronic or magnetic toys your child has and wonder how long it will be before that $30 disk gets carried across the room in the magnetic grasp of R2D2 (batteries not included). My point here is, obviously, that educational software companies should have decent warranties. Free replacement of defective materials for 60 to 90 days is not an uncommon warranty policy. After that, you will typically have to pay five to ten dollars for a replacement copy.

The best warranty policy around is offered by MECC (The Minnesota Educational Computing Consortium). MECC was established in 1973 to assist Minnesota schools and colleges in implementing educational computing. Besides distributing software, MECC trains educators, publishes a newsletter, and provides administrative software to schools. This or-

ganization is attuned to the needs of computer-using educators who, believe it or not, have much less money to spend on software than you or I. In the classroom, a couple of crashed diskettes can virtually end the computer-assisted portion of a curriculum. MECC's warranty is simple and unusual—each courseware product is unconditionally guaranteed. If the original disk dies, no matter how horribly, just return it to MECC for a replacement. During the return period, use the backup disk which comes with each software package. (No, it was not a free copy to give to a friend.) At $44 per package, the price is not out of line with other educational software.

A different, and equally rare, approach these days is for a company not to rely on a warranty policy. There are a few software companies that allow and encourage you to make your own backup copies. Be good to these folks (i.e., do not distribute pirated copies); they are rare gems in the deep dark caverns of the computer world.

## *NEXT MONTH*

Down to the nitty-gritty, how do you measure the educational value of "educational" software? We have considered how you can set goals for your child's learning at home. And we've seen how the design of software and documentation can make a program look good—at least on the surface. Next, let's dive into the real analytical stuff. Does the program set out valid educational objectives and effectively realize them?

(If you just have to read more before your next issue of *Ahoy!* arrives, go down to your local bookstore and spend $6.95 on Seymour Papert's *Mindstorms.* Subtitled *Children, Computers, and Powerful Ideas,* it purports to explain the language LOGO. What it really does is discuss a provocative theory of child development and learning.)



*Researcher Mark Rosen of Joyce Hakansson Associates tests CBS games with some real experts—two four-year-olds. Game is* **Ducks Ahoy!** *(wonderful name!)*
*READER SERVICE NO. 188*




***Topics covered by Krell's*** **SAT Preparation Series** ***include vocabulary, word relationships, sentence completion, reading comprehension, math, and written English.*** **READER SERVICE NO. 187**

## Programs for the Commodore 64 mentioned in this article:

*Animated Arithmetic*, PMI, P.O. Box 87, Buckfield, ME, (800) 227-1836. $10.95 tape, for grades K to two.

*Flower Power Math Fun*, Softwave Productions, 156 Drakes Lane, Summertown, TN 38483, (615) 964-3573. $39.95 disk, for ages five-15.

*In Search of the Most Amazing Thing*, Spinnaker Software, 215 First Street, Cambridge, MA 02142, (617) 868-4700. $44.95 disk, for grades five and up.

*KinderComp*, Spinnaker Software. $29.95 disk or cartridge, for grades pre-K to three.

*Letter Games*, Island Software, Box 300, Dept. B, Lake Grove, NY 11755. $20 tape, $25 disk, for grades pre-K to three.

*MasterType*, Lightening Software, P.O. Box 11725, Palo Alto, CA 94306, (415) 327-3280. $39.95 disk.

*SAT Preparation Series*, Krell Software, 1320 Stony Brook Road, Stony Brook, NY 11790. (516) 751-5139. $299.95 disks, for high school.

*Spellbound*, Timeworks, Inc., P.O. Box 321, Deerfield, IL 60015, (312) 291-9200. $24.95, for ages six to eighteen.

*Word Factory*, Island Software. $20 tape, $25 disk, for grades K and up.

*Milliken Publishing Company*, 1100 Research Boulevard, St. Louis, MO 63132, (314) 991-4220.
*Frenzy*, $39.95 cartridge, for grades two to six.
*Gulp!!*, $39.95 cartridge, for grades two to six.
*Golf Classic*, $39.95 cartridge, for grades four to twelve.

Minnesota Educational Computing Consortium, 3490 Lexington Avenue North, St. Paul, MN 55112, (612) 481-3527 (orders and catalog), (612) 481-3612 (newsletter).
*Metric Mathematics*, $44 disk, for grades four to six.
*Basic Arithmetic*, $44 disk, for grades two to six.
*Elementary Math Logic*, $44 disk, for grades one to six.
*Geography*, $44 disk, for grades four to ten.
*Word Games*, $44 disk, for grades one to six.

# FUTURE OF COMMODORE?

*Continued from page 48*

### *ROOM AT THE TOP*

Although it currently has no computer models in the over-$1000 price range, Commodore has not dismissed that market completely. One sure indication of their interest has been the licensing of manufacturing rights to the 16-bit Z8000 microprocessor, and the acquisition of facilities to produce the RAM chips needed for the memory-hungry machines they go in. Since Commodore's strategy for maintaining low prices has always included manufacturing all vital components themselves, this can be seen as an important step toward their first 16-bit machine. It has also been reported that Commodore has licensed AT&T's popular Unix operating system, which is used to control large multiuser mainframe applications. Although Commodore has not said much about a Z8000 machine, an announcement may be forthcoming as early as this month (April) at the Hanover Fair in Germany. And though any 16-bit machine likely to appear now would almost certainly be aimed at the business market, the experience in the field may prove crucial to Commodore when the more powerful technology becomes inexpensive enough to move into the home market.

### *NEW PIES TO FINGER*

In the future, you can also look for Commodore to branch out into new computer-related fields. Besides manufacturing their own computer components, they are currently negotiating a joint venture project with a Japanese company to manufacture their own disk drives. They have already entered the software business in a big way, recently forming Commodore Software as a separate division. It is certain that their involvement in software will become even greater in the days ahead, as they work to develop more software themselves, and move ahead with their aggressive acquisition of marketing rights to popular titles from outside developers. For the time being Commodore is only selling software for its own brand of computers, but it should not be too long until they begin to translate their most popular titles for computers such as the Apple, Atari, and IBM PC. When Commodore devotes the full force of its enormous marketing power to the task of bringing out quality software at the lowest possible price, it could well revolutionize the pricing structure of the entire home computer software industry.

Another area where you can look to more from Commodore in the future is in the field of computer publications. Their two popular magazines, *Power/Play* and *Commodore: The Microcomputer Magazine,* are likely to increase both in size and frequency of publication in the near future.

Commodore's future is far from the gloomy portrait painted by some recent accounts. Despite any temporary confusion caused by management changes, the company has a tremendous amount of momentum going for it. It has had more than its share of internal turmoil in the past, but its cost-effective manufacturing structure has always been the root of its strength, and will continue to see it through any difficulties. As sales manager Havlik said, "I've been with Commodore for four years now, and I see only good things to come. I can't be too specific about our plans, but you won't be disappointed if you expect great things from Commodore."□

# PROGRAM LISTINGS

On the following pages are listed several programs that we hope you'll want to punch in your Commodore computer. But please read the following introduction first; there are a few things you'll need to know.

Certain computer commands are displayed on the monitor by a variety of odd-looking characters. To get your computer to display these commands rather than actually perform them, you'll need to enter the quote mode. Hold down the SHIFT key and press the "2" key; a set of quote marks will appear. This tells the computer that the characters that follow are to be displayed, not performed. To exit the quote mode, type another set of quote marks, or hit the RETURN key. You'll also enter the quote mode when you INserT spaces or characters onto a line.

In *Ahoy!'s* program listings, you'll frequently find letters and/or numbers surrounded by brackets { }. That's because, for the purposes of clear reproduction, we at *Ahoy!* use a daisy wheel printer incapable of reproducing command symbols. For example, when you're in the quote mode and press the SHIFT and CLR/HOME keys at the same time, the screen (or a dot-matrix printer) will indicate this command with a heart {♥}. Because a daisy wheel cannot duplicate this symbol, it substitutes an alternate code between brackets. In the case of the SHIFT/CLR HOME symbol, our printer substitutes {SC}.

Another special case is SHIFT and COMMODORE characters. We represent these by underlining or overlining, respectively: any character underlined in the program listing should be punched in as a SHIFTed character ( $\underline{J}$ = SHIFT J), any character overlined should be punched in as a COMMODORE character ( $\overline{J}$ = COMMODORE J).

An alternate way of entering commands and other graphics symbols and characters is to use their corresponding character strings. The CLR/HOME command, for example, is entered by typing CHR$(147). While this requires a few extra strokes, it facilitates editing your program or reading the printed listing. For a complete list of CHR$ codes, consult the appendix at the back of your Commodore user manual.

Below is a list of the command abbreviations you'll find in our program listings, the commands they stand for, how to enter them, and how they'll appear on the screen or on a dot matrix printout.

| When You See | It Means | You Type | You Will See | When You See | It Means | You Type | You Will See |
|---|---|---|---|---|---|---|---|
| {SC} | Screen Clear | SHIFT CLR/HOME | | {YL} | Yellow | CNTRL 8 | |
| {HM} | Home | CLR/HOME | | {OR} | Orange | COMMODORE 1 | |
| {CU} | Cursor Up | SHIFT ↑ CRSR ↓ | | {BR} | Brown | COMMODORE 2 | |
| {CD} | Cursor Down | ↑ CRSR ↓ | | {LR} | Light Red | COMMODORE 3 | |
| {CL} | Cursor Left | SHIFT ← CRSR → | | {G1} | Grey 1 | COMMODORE 4 | |
| {CR} | Cursor Right | ← CRSR → | | {G2} | Grey 2 | COMMODORE 5 | |
| {SS} | Shifted Space | SHIFT space | | {LG} | Light Green | COMMODORE 6 | |
| {IN} | Insert | INST | | {LB} | Light Blue | COMMODORE 7 | |
| {RV} | Reverse On | CNTRL 9 | | {G3} | Grey 3 | COMMODORE 8 | |
| {RO} | Reverse Off | CNTRL 0 | | {F1} | Function 1 | F 1 | |
| {BK} | Black | CNTRL 1 | | {F2} | Function 2 | F 2 | |
| {WH} | White | CNTRL 2 | | {F3} | Function 3 | F 3 | |
| {RD} | Red | CNTRL 3 | | {F4} | Function 4 | F 4 | |
| {CY} | Cyan | CNTRL 4 | | {F5} | Function 5 | F 5 | |
| {PU} | Purple | CNTRL 5 | | {F6} | Function 6 | F 6 | |
| {GN} | Green | CNTRL 6 | | {F7} | Function 7 | F 7 | |
| {BL} | Blue | CNTRL 7 | | {F8} | Function 8 | F 8 | |

# C-64 BUG REPELLENT



***By Michael Kleinert and David Barron***

If you're an *Ahoy!* reader, chances are that from time to time you type in programs listed in this section. Unfortunately, man is not perfect and you've probably made errors typing them in. Another sad but true fact is that you then had to check the nonworking programs line by line, hoping to spot your errors. This has never been fun, and never will be.

In order to alleviate this situation, we've developed a system for the Commodore 64 that allows for a quick check of your program. Each line is represented by two letters. You check your code against the code lsted in *Ahoy!*, and if there is a difference, an error exists in that line. This system allows for a fast and easy error check before running the program.

Following this introduction is the program we're speaking of, titled *C-64 Bug Repellent*. It is a basic loader for an assembly language program that resides at $C000 or 49152 (all of you non-assemblers stop worrying and keep reading! Knowledge of assembly language is not necessary.) Before you start to type in a program from *Ahoy!*, run *Bug Repellent*. It will take care of what it has to and then disappear into a cloud of floppy disks. Once it's run, type NEW and proceed to type in the *Ahoy!* program you've selected. When that's done, save your program (do not run it!) and type:

```
SYS 49152 [RETURN]
```

You will be asked if you want the line value codes displayed on the screen or dumped to the printer. If you select printer, make sure it's on. A table of codes will be sent to your printer. If you select screen, the table will appear there. This table will move quickly, too quickly for most mortals. In order to satisfy its human operators the program allows you to pause the listing. This is done by depressing and holding the SHIFT key. To pause for an extended period of time, depress SHIFT LOCK. As long as it is locked, the display will remain frozen.

Compare the table your machine generates to the table in *Ahoy!* that follows the program you're entering. If you spot a difference, jot down the number of the line where the contradiction occurs. When you're done with the comparison, go back and LIST each line where an error occurred, spot the error, and correct it.

Another nifty feature of the program is its ability to inform you as to how many lines your program is. This is another quick way to check whether you entered every line or missed a few.

We hope our program makes computing a bit more enjoyable for you. Don't put up with disgusting bugs in your programs any longer!

(*Important!* Using the VIC 20 Bug Repellent found on the following page requires a different procedure. Read the introduction on page 37 before typing in the program.)

**IMPORTANT**

**Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.**

```
900 FORX= 49152 TO 49445 :READY:P
OKEX,Y:NEXT:END
901 DATA 32, 161, 192, 165, 43, 1
33, 251, 165, 44, 133
902 DATA 252, 160, 0, 230, 251, 2
08, 2, 230, 252, 177
903 DATA 251, 208, 3, 76, 138, 19
2, 230, 251, 208, 2
904 DATA 230, 252, 76, 43, 192, 7
6, 73, 78, 69, 32
905 DATA 35, 32, 0, 169, 35, 160,
 192, 32, 30, 171
906 DATA 160, 0, 177, 251, 170, 2
30, 251, 208, 2, 230
907 DATA 252, 177, 251, 32, 205,
189, 169, 58, 32, 210
908 DATA 255, 169, 0, 133, 253, 2
30, 251, 208, 2, 230
909 DATA 252, 165, 253, 160, 0, 7
6, 16, 193, 133, 253
910 DATA 177, 251, 208, 237, 165,
 253, 41, 240, 74, 74
911 DATA 74, 74, 24, 105, 65, 32,
 210, 255, 165, 253
912 DATA 41, 15, 24, 105, 65, 32,
 210, 255, 169, 13
```

```
913 DATA 32, 220, 192, 230, 63, 2
08, 2, 230, 64, 230
914 DATA 251, 208, 2, 230, 252, 7
6, 11, 192, 169, 153
915 DATA 160, 192, 32, 30, 171, 1
66, 63, 165, 64, 76
916 DATA 231, 192, 96, 76, 73, 78
, 69, 83, 58, 32
917 DATA 0, 169, 247, 160, 192, 3
2, 30, 171, 169, 3
918 DATA 133, 254, 32, 228, 255,
201, 83, 240, 6, 201
919 DATA 80, 208, 245, 230, 254,
32, 210, 255, 169, 4
920 DATA 166, 254, 160, 255, 32,
186, 255, 169, 0, 133
921 DATA 63, 133, 64, 32, 189, 25
5, 32, 192, 255, 166
922 DATA 254, 32, 201, 255, 169,
13, 32, 210, 255, 96
923 DATA 32, 210, 255, 173, 141,
2, 41, 1, 208, 249
924 DATA 96, 32, 205, 189, 169, 1
3, 32, 210, 255, 32
925 DATA 204, 255, 169, 4, 76, 19
5, 255. 147, 83, 67
926 DATA 82, 69, 69, 78, 32, 79,
82, 32, 80, 82
927 DATA 73, 78, 84, 69, 82, 32,
63, 32, 0, 0
928 DATA 0, 0, 170, 177, 251, 201
, 32, 240, 6, 138
929 DATA 24, 113, 251, 10, 170, 1
38, 76, 88, 192, 0
930 DATA 0, 0, 0, 0
```

### BUG REPELLENT LINE CODES FOR C-64 BUG REPELLENT

```
LINE # 900:HA     LINE # 913:PI     LINE # 926:EI
LINE # 901:PE     LINE # 914:EE     LINE # 927:KA
LINE # 902:CE     LINE # 915:FA     LINE # 928:II
LINE # 903:CI     LINE # 916:GA     LINE # 929:KA
LINE # 904:MA     LINE # 917:LM     LINE # 930:KA
LINE # 905:IM     LINE # 918:GE     LINES: 31
LINE # 906:HI     LINE # 919:MA
LINE # 907:OI     LINE # 920:PE
LINE # 908:HI     LINE # 921:NI
LINE # 909:FE     LINE # 922:EA
LINE # 910:II     LINE # 923:CE
LINE # 911:FE     LINE # 924:AA
LINE # 912:DE     LINE # 925:KM
```

# VIC 20 BUG REPELLENT

**From page 38**

```
63000 FORX = 828 TO 1020 :READY:P
OKEX,Y:NEXT:END
63001 DATA 169, 0, 133, 63, 133,
64, 165, 43, 133, 251
63002 DATA 165, 44, 133, 252, 160
, 0, 230, 251, 208, 2
63003 DATA 230, 252, 177, 251, 20
8, 3, 76, 208, 3, 230
63004 DATA 251, 208, 2, 230, 252,
 169, 244, 160, 3, 32
63005 DATA 30, 203, 160, 0, 177,
251, 170, 230, 251, 208
63006 DATA 2, 230, 252, 177, 251,
 32, 205, 221, 169, 58
63007 DATA 32, 210, 255, 169, 0,
133, 253, 230, 251, 208
63008 DATA 2, 230, 252, 165, 253,
 160, 0, 170, 177, 251
63009 DATA 201, 32, 240, 6, 138,
24, 113, 251, 10, 170
63010 DATA 138, 133, 253, 177, 25
1, 208, 226, 165, 253, 41
63011 DATA 240, 74, 74, 74, 74, 2
4, 105, 65, 32, 210
63012 DATA 255, 165, 253, 41, 15,
 24, 105, 65, 32, 210
63013 DATA 255, 169, 13, 32, 210,
 255, 173, 141, 2, 41
63014 DATA 1, 208, 249, 230, 63,
208, 2, 230, 64, 230
63015 DATA 251, 208, 2, 230, 252,
76, 74, 3, 169, 236
63016 DATA 160, 3, 32, 30, 203, 1
66, 63, 165, 64, 32
63017 DATA 205, 221, 169, 13, 32,
 210, 255, 32, 204, 255
63018 DATA 169, 4, 32, 195, 255,
96, 76, 73, 78, 69
63019 DATA 83, 58, 32, 0, 76, 73,
 78, 69, 32, 35
63020 DATA 32, 0, 0
```

## BUG REPELLENT LINE CODES FOR VIC 20 BUG REPELLENT

```
LINE # 63000:HA     LINE # 63011:OI
LINE # 63001:EM     LINE # 63012:OI
LINE # 63002:CI     LINE # 63013:OE
LINE # 63003:LI     LINE # 63014:PI
LINE # 63004:IA     LINE # 63015:FA
LINE # 63005:MA     LINE # 63016:CA
LINE # 63006:GI     LINE # 63017:BM
LINE # 63007:MA     LINE # 63018:BE
LINE # 63008:EM     LINE # 63019:CM
LINE # 63009:II     LINE # 63020:KA
LINE # 63010:ME     LINES: 21
```

# VIC CALCULATOR

From page 17

**IMPORTANT**

**Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.**

```
•10 REM VIC CALCULATOR
•20 REM J.WOYTON 12/83
•30 S=1:O$="+"
•40 PRINTCHR$(147)"OP  CALC";TAB(1
 3)CHR$(18)"MEMORY"CHR$(146):PRINT
•45 IFABS(A)<.01THENA=0
•46 IFABS(M)<.01THENM=0
•47 A=VAL(LEFT$(STR$(A),9)):M=VAL(
 LEFT$(STR$(M),9))
•50 PRINT"="A:IFM<>0THENPRINTTAB(1
 1)CHR$(145)CHR$(18)MCHR$(146)
•60 PRINTO$;TAB(3)S$+N$
•70 GETI$:IFI$=""THENI=0:GOTO70
•80 IFI$="="THENI$="+"
•90 IFI$="+"THENI=1
•100 IFI$="-"THENI=2
•110 IFI$="*"THENI=3
•120 IFI$="/"THENI=4
•130 IFI$=CHR$(133)THENI=5
•140 IFI$=CHR$(134)THENI=6
•150 IFI$=CHR$(135)THENI=7
•160 IFI$=CHR$(139)THENI=8
•170 IFI$=CHR$(140)THENI=9
•180 IFI$=CHR$(136)THENI=10
•190 IFI$=CHR$(32)THENS=-S:I$="":I
 =11
•195 GOTO490
•200 ONIGOSUB260,260,260,260,380,3
 90,400,420,430,440
•210 IFS<0THENS$="-"
•220 IFS>0THENS$=" "
•230 N$=N$+I$:N$=LEFT$(N$,8)
•240 PRINTCHR$(145)CHR$(145)CHR$(1
 45)
•250 GOTO45
•260 IFN$=""THENO$=I$:I$="":RETURN
•270 IFO$="+"THENO=1
•280 IFO$="-"THENO=2
•290 IFO$="*"THENO=3
•300 IFO$="/"THENO=4
•310 N=VAL(S$+N$):N$="":S=1:O$=I$:
 I$=""
•320 PRINTCHR$(17)
•330 ONOGOTO340,350,360,370
•340 A=A+N:RETURN
•350 A=A-N:RETURN
•360 A=A*N:RETURN
•370 A=A/N:RETURN
•380 M=M+A:M$="M+":GOTO450
•390 M=M-A:M$="M-":GOTO450
•400 IFM<0THENS=-S
•410 N$=STR$(ABS(M)):N$=MID$(N$,2,
 8):M$="MR":GOTO450
•420 M=0:M$="MC":GOTO450
•430 A=0:N$="":M$="CC":S=1:O$="+":
 GOTO450
•440 N$="":S=1:M$="CE"
•450 PRINTCHR$(145)"
       "
•460 PRINTCHR$(145)M$CHR$(17)CHR$(
 17)
•470 I$=""
•480 RETURN
•490 FORK=48TO57
•500 IFI$=CHR$(K)THENI=11
•510 NEXTK
•520 IFI$=CHR$(46)THENI=11
•530 IFI=0THEN70
•540 GOTO200
```

## BUG REPELLENT LINE CODES FOR VIC CALCULATOR

```
LINE # 10:KA        LINE # 30:MA
LINE # 20:PM        LINE # 40:AE
```

```
LINE # 45:IA
LINE # 46:EA
LINE # 47:LM
LINE # 50:DE
LINE # 60:GA
LINE # 70:EI
LINE # 80:MA
LINE # 90:ME
LINE # 100:MI
LINE # 110:EM
LINE # 120:NA
LINE # 130:JE
LINE # 140:BI
LINE # 150:JM
LINE # 160:KA
LINE # 170:CE
LINE # 180:MI
LINE # 190:AM
LINE # 195:OI
LINE # 200:KA
LINE # 210:NA
LINE # 220:HI
LINE # 230:CE
LINE # 240:CM
LINE # 250:AE
LINE # 260:OI
LINE # 270:CE
LINE # 280:CI
LINE # 290:KM
LINE # 300:DA
LINE # 310:HI
LINE # 320:KM
LINE # 330:CI
LINE # 340:GI
LINE # 350:II
LINE # 360:KI
LINE # 370:MI
LINE # 380:EI
LINE # 390:EI
LINE # 400:OE
LINE # 410:EI
LINE # 420:EI
LINE # 430:EI
LINE # 440:KA
LINE # 450:EI
LINE # 460:KM
LINE # 470:HI
LINE # 480:DI
LINE # 490:ME
LINE # 500:MM
LINE # 510:DM
LINE # 520:MM
LINE # 530:OI
LINE # 540:IA
LINES: 58
```

# MEMORY MANAGEMENT

From page 33

## TEST 1

```
•10 REM:MEMORY MANAEMENT TEST 1
•20 IFPEEK(251)=GOTO90
•30 PRINT"TEST 1"
•40 A$="HELLO FROM TEST 1"
•50 GOSUB20:GOSUB300
•60 POKE49152,0:REM FOR VIC POKE 1
024,0
•70 POKE44,192:REM FDATA VIC PDATA
44,4
•80 LOAD"TEST 2",8
•90 PRINTB$:GOSUB300
•110 PRINT"BACK IN TEST 1"
•120 GOSUB200:END
•200 PRINTPEEK(51);PEEK(52);PEEK(5
5);PEEK(56)
•210 PRINTPEEK(47);PEEK(48);PEEK(4
9);PEEK(50)
•220 RETURN
•300 PRINT"HIT A KEY"
•310 GET C$:IF C$=""THEN310
•320 RETURN
```

## TEST 2

```
•10 REM:MEMORY MANAGEMENT TEST 2
•20 PRINTA$
•30 PRINT"NOW IN TEST 2"
•40 B$="GREETINGS FROM TEST 2"
•50 GOSUB200:GOSUB300
•60 POKE251,1
•70 POKE44,8:REM FOR VIC POKE 44,1
8
•80 GOTO10
•200 PRINTPEEK(43);PEEK(44);PEEK(4
5);PEEK(46)
•210 PRINTPEEK(47);PEEK(48);PEEK(4
9);PEEK(50)
•220 PRINTPEEK(51);PEEK(52);PEEK(5
5);PEEK(56)
•230 RETURN
•300 PRINT"HIT A KEY"
•310 GET C$:IF C$=""THEN310
•320 RETURN
```

## BUG REPELLENT LINE CODES FOR MEMORY MANAGEMENT

### TEST 1

```
LINE # 10:DE
LINE # 20:BI
LINE # 30:PA
LINE # 40:PA
LINE # 50:JA
LINE # 60:CA
LINE # 70:LA
LINE # 80:CA
LINE # 90:JA
LINE # 110:PA
LINE # 120:NA
LINE # 200:KE
LINE # 210:HE
LINE # 220:DI
LINE # 300:EA
LINE # 310:NI
LINE # 320:DI
LINES: 17
```

### TEST 2

```
LINE # 10:DI
LINE # 20:CI
LINE # 30:PI
LINE # 40:PI
LINE # 50:JA
LINE # 60:NE
LINE # 70:CI
LINE # 80:NI
LINE # 200:JE
LINE # 210:HE
LINE # 220:KE
LINE # 230:DI
LINE # 300:EA
LINE # 310:NI
LINE # 320:DI
LINES: 15
```

# I GOT BIORHYTHM

From page 64

**IMPORTANT**
**Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.**

```
•2 PRINT"{SC}{WH}"
•4 V=53280:FORT=0TO30STEP.25:A=INT
(16+15*SIN(T)):PRINTTAB(A);:PRINT
"{WH}BIORHYTHM"
•6 C=INT(16*RND(1))+1:POKEV+1,C:NE
XT
•8 POKEV,6:POKEV+1,8
•10 PRINT"{WH}{SC}{CD}{CD}{CD}{CD}
{CD}{CD}"SPC(14);"{RV} BIORHYTHM
"
•12 PRINT"{CD}{CD}"SPC(9);"A PROGR
AM FOR THE C-64"
•14 PRINT"{CD}"SPC(13);"BY B.W.BEH
LING"
•16 FORI=0TO2000:NEXT
•18 DIMM(12):M(1)=31:M(2)=28:M(3)=
31:M(4)=30:M(5)=31:M(6)=30:M(7)=3
1:M(8)=31
•20 M(9)=30:M(10)=31:M(11)=30:M(12
)=31:M(0)=0
•22 PRINT"{SC}{WH}{CD}{CD}{CD}
 ENTER THE BIRTHDATE IN NUMBERS."
•24 PRINT"{CD}{CD}{CD} EXAMPLE:"
•26 PRINT"{CD}           APRIL 9, 196
3 = 04/09/1963"
•28 INPUT"{CD}{CD}{CD}{CD} BIRTHDA
TE";BI$
•30 REM EMPTY LINE FOR FREQUENT DA
TES
•32 REM EMPTY LINE FOR FREQUENT DA
TES
•34 REM EMPTY LINE FOR FREQUENT DA
TES
•36 REM EMPTY LINE FOR FREQUENT DA
TES
•38 REM EMPTY LINE FOR FREQUENT DA
TES
•40 PRINT"{SC}{CD}{CD}{CD}
 ENTER DATE FOR BIORHYTHM."
•44 INPUT"{CD}{CD}{CD}{CD} DATE";D
A$
•45 IFLEN(BI$)<>10ORLEN(DA$)<>10TH
ENGOTO250
•46 BM$=LEFT$(BI$,2):BD$=MID$(BI$,
4,2):BY$=RIGHT$(BI$,4)
•47 BM%=VAL(BM$)-1:BD%=VAL(BD$):BY
%=VAL(BY$)
•48 DM$=LEFT$(DA$,2):DD$=MID$(DA$,
4,2):DY$=RIGHT$(DA$,4)
•49 DM%=VAL(DM$)-1:DD%=VAL(DD$):DY
%=VAL(DY$)
•50 BL=BY%/4:IFINT(BL)=BLTHENBL=1:
M(2)=29:GOTO54
•52 BL=0:M(2)=28
•54 BT%=BD%
•56 FORI=0TOBM%:BT%=BT%+M(I):NEXT
•60 DL=DY%/4:IFINT(DL)=DLTHENDL=1:
M(2)=29:GOTO64
•62 DL=0:M(2)=28
•64 DT%=DD%
•66 FORI=0TODM%:DT%=DT%+M(I):NEXT
•70 IFBY%=DY%THENTD%=DT%-BT%:GOTO9
0
•72 IFBL=1THENTD%=366-BT%:GOTO76
•74 TD%=365-BT%
•76 YT%=(DY%-BY%)-1
•78 TD%=TD%+(INT((YT%*365.25))+DT%
)
•90 PH=TD%/23:PH=(PH-INT(PH))*(2*3
.1416):PH=SIN(PH)
•92 EM=TD%/28:EM=(EM-INT(EM))*(2*3
.1416):EM=SIN(EM):IFEM<-.005OREM>
.005THENGOTO94
•93 EM=0
•94 IN=TD%/33:IN=(IN-INT(IN))*(2*3
.1416):IN=SIN(IN)
•96 PH$=STR$(PH):EM$=STR$(EM):IN$=
STR$(IN):PH$=LEFT$(PH$,4):EM$=LEF
T$(EM$,4)
•98 IN$=LEFT$(IN$,4)
•100 PRINT"{SC}{CD} YOU HAVE BEEN
ALIVE"TD%;"DAYS!!"
•102 PRINT"{CD}{CD} YOUR PHYSICAL
CYCLE VALUE IS  "PH$
•104 PRINT"{CD}{CD} YOUR EMOTIONAL
 CYCLE VALUE IS  "EM$
•106 PRINT"{CD}{CD} YOUR INTELLECT
UAL CYCLE VALUE IS  "IN$
•108 PRINT"{CD}{CD}{CD} A VALUE CL
OSE TO 1. IS AN 'UP' VALUE."
•110 PRINT"{CD} A VALUE CLOSE TO -
1. IS A 'DOWN' VALUE."
```

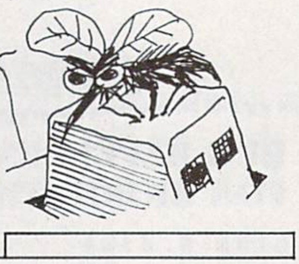
```
•112 PRINT" A VALUE OF 0 IS CONSID
 ERED CRITICAL!!"
•114 PRINT"{CD}{CD} HIT [N] FOR AN
 OTHER DATE."
•118 PRINT"{CD} HIT SPACE FOR 7 DA
 Y FORECAST."
•120 GETK$:IFK$=""THEN120
•122 IFK$="N"THENRUN18
•130 PRINT"{SC} DAY","PHYSI.","EMO
 TI.","INTEL."
•132 PRINT" {RV}"TD%,PH$,EM$,IN$
•134 FORI=1TO7:TD%=TD%+1
•136 PH=TD%/23:PH=(PH-INT(PH))*(2*
 3.1416):PH=SIN(PH)
•138 EM=TD%/28:EM=(EM-INT(EM))*(2*
 3.1416):EM=SIN(EM):IFEM<-.005OREM
 >.005THENGOTO140
•139 EM=0
•140 IN=TD%/33:IN=(IN-INT(IN))*(2*
 3.1416):IN=SIN(IN)
•142 PH$=STR$(PH):EM$=STR$(EM):IN$
 =STR$(IN):PH$=LEFT$(PH$,4):EM$=LE
 FT$(EM$,4)
•144 IN$=LEFT$(IN$,4)
•146 PRINT"{CD} "TD%,PH$,EM$,IN$
•148 NEXT
•150 PRINTSPC(14);"{CD}**COMMANDS*
 *"
•151 PRINT"{CD} [N] = NEW DATE."
•152 PRINT" [E] = END PROGRAM."
•154 PRINT" [C] = TO CONVERT DAY #
  TO DATE."
•155 PRINT" SPACE = CONTINUE TO NE
 XT 7 DAYS."
•162 GETK$:IFK$=""THEN162
•164 IFK$="N"THENRUN18
•166 IFK$="C"THEN200
•168 IFK$="E"THENEND
•170 GOTO130
•200 INPUT"{SC}{CD}{CD}{CD} ENTER
 DAY # TO BE CONVERTED";DC%
•205 TD%=TD%-7:IFTD%<=DC%THENGOTO2
 20
•210 PRINT"{CD}{CD}{CD} INVALID DA
 Y # !!!"
•215 PRINT"{CD}{CD} DAY # MUST BE
 GREATER THAN";TD%;"!!":FORI=0TO35
 00:NEXT:GOTO200
•220 CT%=DC%-TD%:CM%=DM%+1:CD%=DD%
 :CY%=DY%
•225 IFCT%>(M(CM%)-CD%)THENCT%=CT%
 -(M(CM%)-CD%):CM%=CM%+1:CD%=0
•230 IFCM%>12THENCM%=0:CY%=CY%+1:M
 (2)=28:IFCY%/4=INT(CY%/4)THENM(2)
 =29
•235 IFCT%<=(M(CM%)-CD%)THENCD%=CD
 %+CT%:GOTO245
•240 GOTO225
•245 PRINT"{CD}{CD}{CD} DATE:"CM%;
 "/";CD%;"/";CY%
•250 PRINT"{CD}{CD}{CD} HIT SPACE
 TO RETURN TO 7 DAY FORECAST."
•255 PRINT"{CD} HIT [C] TO CONVERT
  ANOTHER DAY."
•260 GETK$:IFK$=""THEN260
•265 IFK$="C"THENTD%=TD%+7:GOTO200
•270 GOTO130
•275 PRINT"{CD}{CD}{CD}"
•280 FORI=1TO5:PRINT" ERROR!!";:NE
 XT
•285 PRINT"{CD} YOU MUST ENTER THE
  COMPLETE DATE!!!"
•290 PRINT"{CD} ALL ZEROS MUST BE
 INCLUDED!!!"
•300 PRINT"{CD}{CD}{CD} HIT SPACE
 TO RESET."
•305 GETK$:IFK$=""THEN305
•310 RUN18
```



## BUG REPELLENT LINE CODES FOR BIORHYTHM

| | |
|---|---|
| LINE # 2:GA | LINE # 40:EI |
| LINE # 4:DA | LINE # 44:DI |
| LINE # 6:MI | LINE # 45:GI |
| LINE # 8:JA | LINE # 46:ME |
| LINE # 10:DA | LINE # 47:BE |
| LINE # 12:GI | LINE # 48:ME |
| LINE # 14:MA | LINE # 49:FE |
| LINE # 16:NI | LINE # 50:II |
| LINE # 18:JM | LINE # 52:DA |
| LINE # 20:GA | LINE # 54:FE |
| LINE # 22:KI | LINE # 56:AI |
| LINE # 24:KI | LINE # 60:JA |
| LINE # 26:GA | LINE # 62:DA |
| LINE # 28:FI | LINE # 64:HE |
| LINE # 30:NE | LINE # 66:AI |
| LINE # 32:NE | LINE # 70:JI |
| LINE # 34:NE | LINE # 72:KA |
| LINE # 36:NE | LINE # 74:PE |
| LINE # 38:NE | LINE # 76:IM |

```
LINE # 78:IM
LINE # 90:KE
LINE # 92:EI
LINE # 93:MA
LINE # 94:GE
LINE # 96:EE
LINE # 98:IE
LINE # 100:MA
LINE # 102:FA
LINE # 104:MI
LINE # 106:BA
LINE # 108:GI
LINE # 110:GI
LINE # 112:OA
LINE # 114:AI
LINE # 118:FI
LINE # 120:MA
LINE # 122:GI
LINE # 130:FI
LINE # 132:JA
LINE # 134:OE
LINE # 136:KE
LINE # 138:NA
LINE # 139:MA
LINE # 140:GE
LINE # 142:EE
LINE # 144:IE
LINE # 146:JA
LINE # 148:AI
LINE # 150:NI
LINE # 151:AI
LINE # 152:OI
LINE # 154:AI
LINE # 155:II
LINE # 162:OI
LINE # 164:GI
LINE # 166:EA
LINE # 168:HI
LINE # 170:II
LINE # 200:MM
LINE # 205:NA
LINE # 210:AA
LINE # 215:AA
LINE # 220:JM
LINE # 225:OA
LINE # 230:DE
LINE # 235:LE
LINE # 240:KE
LINE # 245:OM
LINE # 250:FI
LINE # 255:KI
LINE # 260:PA
LINE # 265:IA
LINE # 270:II
LINE # 275:EA
LINE # 280:II
LINE # 285:AA
LINE # 290:EA
LINE # 300:BI
LINE # 305:OE
LINE # 310:AI
LINES: 99
```



# AIR ASSAULT

**From page 71**

**IMPORTANT**

**Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.**

```
•100 REM ** A I R  A S S A U L T
 **
•110 REM ** DESIGNED BY:BOB LLORET
 **
•120 REM ***  FOR AHOY MAGAZINES
 ***
•130 REM ------------------------
----
•160 PRINT "{SC}":POKE 53280,0:POK
E 53281,4
•170 PRINT TAB(17);"{YL}{CD}{CD}{C
D}{CD}{CD}A I R"
•180 PRINT TAB(13);"{CD}{CD}A S S
A U L T"
•190 PRINT TAB(14);"{BL}{CD}{CD}{C
D}{CD}{CD}{CD}DESIGNED BY:"
•200 PRINT TAB(15);"{CD}BOB LLORET
"
•210 PRINT TAB(10);"{BK}{CD}{CD}{C
D}{CD}PRESS [F1] TO START"
•220 GET A$:IF A$="{F1}" THEN 230
•225 GOTO 220
•230 PRINT "{SC}":POKE 53280,0:POK
E 53281,0:CLR:C=C+1
•240 PRINT "{LB}{CD}{CD}{CD}{CD}{C
D}{CD}  HOLD ON....I AM BUILDIN
G THE CITY.":IF C>1 THEN 260
•250 GOTO 700
•260 GOSUB 590
•270 GOSUB 1000
•280 REM *****  MAIN LOOP  *****
•285 REM =======================
•310 SI=54272:FOR N=0 TO 24:POKE S
I+N,0:NEXT N
•320 POKE SI+24,3:POKESI+5,0:POKES
I+6,255:BOMB=140
•325 Y=40:XPL=1944:HB=56216:X1=130
:Y1=175:POKE V+6,X1:POKE V+7,Y1:P
OKE V+21,8
•326 FOR D=1 TO 1000:NEXT D
•328 IF B>39 THEN POKE V+21,0:GOTO
 1200
•330 X=INT(RND(0)*235)+20:SP=INT(R
ND(0)*5)+6:B=B+1:PRINT TAB(33);"{
WH}{CU}{CU}{CU}{CU}{CU}";B
•335 PRINT TAB(33);"{CD}{CD}{CD}{C
D}";NAMT:PRINT "{CU}{CU}"
•360 JS=PEEK(56321):FI=JSAND16:JS=
15-(JSAND15):BOMB=BOMB-2
•365 IF JS=1 THEN Y1=Y1-7:IF Y1<=1
00 THEN Y1=100
•370 IF JS=2 THEN Y1=Y1+7:IF Y1>=1
90 THEN Y1=190
•375 IF JS=4 THEN X1=X1-7:IF X1<=2
0 THEN X1=20
•380 IF JS=8 THEN X1=X1+7:IF X1>=2
55 THEN X1=255
```

```
•381 POKE V+21,12:POKE V+6,X1:POKE
 V+7,Y1
•385 IF PEEK(V+30)ANDX=X AND FI=0
THEN CH=INT(X/9):RH=INT(Y/11):GOT
O 420
•390 Y=Y+SP:POKE V+4,X:POKE V+5,Y:
POKE SI,BOMB:POKE SI+1,BOMB:POKE
SI+4,17
•400 IF Y>=226 THEN Y=40:HIT=INT(X
/8):POKESI+4,16:POKESI+1,0:POKESI
,0:GOTO 480
•410 GOTO 360
•415 REM ****  SHOT DOWN  ****
•416 REM =====================
•420 POKE SI+4,16:POKE SI+1,0:POKE
 SI,0:POKE SI+5,11:POKE V+21,8:PO
KE 53281,2
•430 YB=1024+CH+40*RH:XB=55296+CH+
40*RH
•435 POKE XB,7:POKE XB+1,7:POKE XB
+40,7:POKE XB+41,7
•440 POKE YB,41:POKE YB+1,42:POKE
YB+40,43:POKE YB+41,44
•450 FOR A=15TO4 STEP-.6:POKESI+24
,A:POKESI+4,129:POKESI+1,3:POKESI
,50:NEXT A
•460 FORD=1 TO 200:NEXT D:POKESI+4
,16:POKESI+1,0:POKESI,0
•465 POKE YB,32:POKE YB+1,32:POKE
YB+40,32:POKE YB+41,32
•470 POKE 53281,6:Y=40:BOMB=140
•472 IF Y<22 OR Y>20 THEN AMT=10
•474 IF Y<19 OR Y>17 THEN AMT=30
•476 IF Y<=16 THEN AMT=75
•477 NAMT=NAMT+AMT:T=T+1:GOTO 328
•478 REM ****  HIT BUILDING  ****
•479 REM ========================
•480 POKE SI+5,11:POKEV+21,8:POKE
53281,2
•490 POKE HB+HIT-2,7:POKE HB+HIT-1
,7:POKE HB+HIT,7
•500 POKE HB+40+HIT-2,7:POKE HB+40
+HIT-1,7:POKE HB+40+HIT,7
•501 FORA=15TO 9STEP-.4:POKESI+24,
A:POKESI+4,129:POKESI+1,1:POKESI,
30:NEXT A
•502 POKE XPL+HIT-2,35:POKE XPL+HI
T-1,36:POKE XPL+HIT,33
•505 POKE XPL+40+HIT-2,38:POKE XPL
+40+HIT-1,39:POKE XPL+40+HIT,40
•510 FOR D=1 TO 200:NEXT D:POKE 53
281,7
•515 POKE HB+HIT-2,1:POKE HB+HIT-1
,1:POKE HB+HIT,1
•520 POKE HB+40+HIT-2,1:POKE HB+40
+HIT-1,1:POKE HB+40+HIT,1:FORD=1T
O200:NEXT
•525 POKE HB+HIT-2,2:POKE HB+HIT-1
,2:POKE HB+HIT,2
•530 POKE HB+40+HIT-2,2:POKE HB+40
+HIT-1,2:POKE HB+40+HIT,2:FORD=1T
O200:NEXT
•535 POKE 53281,6:POKE XPL+HIT-2,3
2:POKE XPL+HIT-1,32:POKE XPL+HIT,
32
•540 POKE XPL+40+HIT-2,32:POKE XPL
+40+HIT-1,32:POKE XPL+40+HIT,32
•550 POKE V+5,Y:BOMB=140:POKESI+4,
16:POKESI+1,0:POKESI,0
•560 FOR A=1984 TO 2023:CK=PEEK(A)
:IF CK=32 THEN NEXT A:POKEV+21,0:
GOTO 1200
•570 GOTO 328
•590 REM ****  MAIN SCREEN  ****
•591 REM =======================
•594 PRINT "{SC}":PRINT TAB(9);"{W
H}A AIR ASSAULT A"
•595 A1=55328:FOR ROW=1 TO 25:FOR
S=A1 TO A1+7:POKE S,0:NEXT S:A1=A
1+40:NEXT ROW
•600 A=1056:FOR ROW=1 TO 25:FOR S=
A TO A+7:POKE S,45:NEXT S:A=A+40:
NEXT ROW
•605 POKE 53280,0:POKE 53281,6
•610 PRINT TAB(33);"{CD}{CD}{CD}{W
H}BOMBS:":PRINT TAB(33);"{WH}{CD}
{CD}{CD}{CD}SCORE:":PRINT
•620 FOR A=56216 TO 56247:C=INT(RN
D(0)*5)+1:POKE A,C:POKE A+40,C:NE
XT A
•640 FOR A=1944 TO 1975:READ BLD:P
OKE A,BLD:NEXT A
•650 FOR A=1984 TO 2015:READ BLD:P
OKE A,BLD:NEXT A
•660 RETURN
•698 REM **** MOVE CHAR. SET ****
•699 REM =======================
•700 PRINT CHR$(142):POKE 52,48:PO
KE 56,48:CLR
•710 POKE 56334,PEEK(56334)AND254
•720 POKE 1,PEEK(1)AND251
•730 FOR I=0 TO 511:POKE I+12288,P
EEK(I+53248):NEXT
•740 POKE 1,PEEK(1)OR4:POKE 56334,
PEEK(56334)OR1
•750 POKE 53272,(PEEK(53272)AND240
```

```
)+12
•758 REM **** REDEFINE CHAR ****
•759 REM =======================
•760 READ LOC:IF LOC=-1 THEN 260
•770 FOR A=0 TO 7:READ NUMB:POKE L
OC+A,NUMB:NEXT A
•780 GOTO 760
•790 DATA 12504,0,8,8,28,62,125,25
5,231
•800 DATA 12512,0,0,0,0,0,255,153,
153
•810 DATA 12520,224,224,224,224,22
4,255,255,231
•820 DATA 12528,56,56,56,56,56,255
,153,153
•825 DATA 12536,3,3,3,3,3,3,255,25
5
•830 DATA 12584,255,255,255,255,15
3,153,255,255
•835 DATA 12568,240,118,60,30,143,
199,227,241
•840 DATA 12576,255,126,60,24,129,
195,232,255
•845 DATA 12552,15,30,60,118,241,2
27,199,143
•850 DATA 12592,118,60,30,143,199,
99,49,56
•855 DATA 12600,255,126,60,24,129,
195,232,231
•860 DATA 12608,30,60,118,241,227,
198,140,28
•870 DATA 12616,100,52,132,194,98,
48,24,0
•880 DATA 12624,64,70,142,156,176,
0,0,31
•890 DATA 12632,32,100,200,128,24,
48,96,192
•900 DATA 12640,192,204,102,99,0,1
28,152,132
•910 DATA 12648,255,255,255,255,25
5,255,255,255,-1
•948 REM ****  BUILDING DATA  ****
•949 REM ==========================
•950 DATA 28,27,30,28,31,29,27,37,
28,30,29,37,37,29,27,27,28,37,30,
30
•960 DATA 31,27,37,29,28,27,30,27,
27,28,37,37
•970 DATA 37,37,37,37,37,37,37,37,
37,37,37,37,37,37,37,37,37,37,37,
37
•980 DATA 37,37,37,37,37,37,37,37,
37,37,37,37
•997 REM
•998 REM **** CREATE SPRITES ****
•999 REM ========================
•1000 V=53248:POKE 2042,13:POKE 20
43,14:POKE V+41,0:POKE V+42,1
•1010 FOR N=0 TO 62:READ Q:POKE 83
2+N,Q:NEXT N
•1020 DATA 0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,12,15,248
•1030 DATA 14,31,252,15,255,254,15
,255,254,14,31,252,12,15,248
•1040 DATA 0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0
•1050 FOR N=0 TO 62:READ Q:POKE 89
6+N,Q:NEXT N
•1060 DATA 0,0,0,0,0,0,0,0,0,0,0,0
,3,128,224,2,0,32,2,8,32,0,8,0,0,
8,0,0,0,0
•1070 DATA 0,227,128,0,0,0,0,8,0,0
,8,0,2,8,32,2,0,32,3,128,224
•1080 DATA 0,0,0,0,0,0,0,0,0,0,0,0
•1100 RETURN
•1200 PRINT "{SC}":POKE 53280,6:PO
KE 53281,15:POKE SI,16:POKE SI+1,
0:POKE SI,0
•1210 PRINT TAB(4);"{BL}{CD}{CD}TH
E FOLLOWING ARE THE RESULTS OF"
•1220 PRINT "  THE ENEMY ASSAULT..
.."
•1225 FOR D=1 TO 2000:NEXT D
•1230 PRINT TAB(1);"{RD}{CD}{CD}{C
D}BOMBS DROPPED...........";B
•1240 PRINT TAB(1);"{CD}BOMBS DEFL
ECTED.........";T:DH=B-T
•1250 PRINT TAB(1);"{CD}DIRECT HIT
S.............";DH
•1260 IF DH>=8 THEN MSG$="TOTAL DE
VESTATION":MG$="TOTAL ANIHILATION
":GOTO 1280
•1270 MSG$="PARTIAL DESTRUCTION":M
G$="FEW CASUALTIES"
•1280 PRINT TAB(1);"{CD}STRUCTURE
DAMAGE...";MSG$
•1290 PRINT TAB(1);"{CD}POPULATION
.........";MG$
•1295 IF NAMT>=HY THEN HY=NAMT
•1300 PRINT TAB(1);"{CD}YOUR SCORE
..............";HY
•1320 PRINT TAB(9);"{PU}{CD}{CD}{C
D}{CD}PRESS FIRE TO TRY AGAIN":CL
R
•1325 WAIT 145,16,16
•1330 POKE V+21,0:GOTO 230
```

## BUG REPELLENT LINE CODES FOR AIR ASSAULT

```
LINE # 100:HI
LINE # 110:NI
LINE # 120:DI
LINE # 130:EM
LINE # 160:EA
LINE # 170:II
LINE # 180:EI
LINE # 190:KI
LINE # 200:HI
LINE # 210:OI
LINE # 220:FI
LINE # 225:JA
LINE # 230:EE
LINE # 240:DA
LINE # 250:NA
LINE # 260:HI
LINE # 270:KA
LINE # 280:FI
LINE # 285:AM
LINE # 310:OI
LINE # 320:LA
LINE # 325:BA
LINE # 326:MA
LINE # 328:MA
LINE # 330:OA
LINE # 335:KA
LINE # 360:KA
LINE # 365:FA
LINE # 370:JI
LINE # 375:BA
LINE # 380:BM
LINE # 381:KM
LINE # 385:LA
LINE # 390:KE
LINE # 400:GA
LINE # 410:MA
LINE # 415:NI
LINE # 416:AM
LINE # 420:DI
LINE # 430:HA
LINE # 435:EM
LINE # 440:FA
LINE # 450:PE
LINE # 460:NA
LINE # 465:EA
LINE # 470:LA
LINE # 472:KI
LINE # 474:LI
LINE # 476:OM
LINE # 477:MA
LINE # 478:JI
LINE # 479:AM
LINE # 480:DI
LINE # 490:JM
LINE # 500:JM
LINE # 501:HE
LINE # 502:OE
LINE # 505:OA
LINE # 510:EM
LINE # 515:IE
LINE # 520:FI
LINE # 525:II
LINE # 530:FI
LINE # 535:OA
LINE # 540:OA
LINE # 550:NA
LINE # 560:MA
LINE # 570:MA
LINE # 590:NI
LINE # 591:AM
LINE # 594:BA
LINE # 595:LE
LINE # 600:LE
LINE # 605:EI
LINE # 610:NE
LINE # 620:BE
LINE # 640:DE
LINE # 650:DE
LINE # 660:DI
LINE # 698:FI
LINE # 699:AM
LINE # 700:EA
LINE # 710:LI
LINE # 720:CM
LINE # 730:GI
LINE # 740:JE
LINE # 750:BA
LINE # 758:NI
LINE # 759:AM
LINE # 760:DA
LINE # 770:LE
LINE # 780:AA
LINE # 790:LM
LINE # 800:EE
LINE # 810:LM
LINE # 820:EE
LINE # 825:NM
LINE # 830:NM
LINE # 835:ME
LINE # 840:BM
LINE # 845:LM
LINE # 850:GA
LINE # 855:PM
LINE # 860:LA
LINE # 870:KA
LINE # 880:BM
LINE # 890:CA
LINE # 900:PA
LINE # 910:MM
LINE # 948:BI
LINE # 949:AM
LINE # 950:NI
LINE # 960:NE
LINE # 970:NE
LINE # 980:NE
LINE # 997:DM
LINE # 998:BI
LINE # 999:AM
LINE # 1000:EE
LINE # 1010:AI
LINE # 1020:OA
LINE # 1030:OA
LINE # 1040:KA
LINE # 1050:AI
LINE # 1060:KA
LINE # 1070:AA
LINE # 1080:KA
LINE # 1100:DI
LINE # 1200:NA
LINE # 1210:AI
LINE # 1220:BI
LINE # 1225:MA
LINE # 1230:EA
LINE # 1240:AI
LINE # 1250:LA
LINE # 1260:AA
LINE # 1270:JA
LINE # 1280:FI
LINE # 1290:HI
LINE # 1295:II
LINE # 1300:BE
LINE # 1320:OA
LINE # 1325:CA
LINE # 1330:BI
LINES: 144
```

# MATH MASTER

From page 71

**IMPORTANT**

**Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.**

```
•10 REM ****** MATH MASTER ******
                ***** BY BOB LLORET
 *****
•15 REM ** FOR  AHOY MAGAZINES **
•20 REM *************************
•90 LEV=0:T=0:PROB=0
•100 PRINT "{SC}":POKE 53280,6:POK
E 53281,0
•110 PRINT "{YL}{RV}":PRINT TAB(11
);"{RV}{CU}ACCCCCCCCCCCCCCCCCS"
•120 PRINT TAB(11);"{RV}B*  MATH M
ASTER  *B"
•130 PRINT TAB(11);"{RV}ZCCCCCCCCC
CCCCCCCCX"
•140 PRINT:PRINT TAB(13);"BY: BOB
```

```
LLORET":PRINT:PRINT
•150 PRINT "{CD}{CD}{YL}EEEEEEEEEE
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE"
•160 PRINT TAB(11);"{CU}{CU}{CU}{C
U}{LB}FIRST NAME:";:INPUT NAME$
•161 IF NAME$="" THEN PRINT "{CD}{
CD}":GOTO 160
•162 FOR A=1 TO 2:PRINT:NEXT A
•165 S=54272:FOR L=0 TO 24:POKE S+
L,0:NEXT
•166 POKE S+1,22:POKE S,96:POKE S+
5,10:POKE S+24,10:POKE S+4,33
•170 PRINT TAB(11);"{GN}[1]..ADDIT
ION":PRINT:PRINT TAB(11);"{PU}[2]
..SUBTRACTION"
•180 PRINT:PRINT TAB(11);"{RD}[3].
.MULTIPLICATION"
•190 PRINT:PRINT TAB(11);"{BR}[4].
.DIVISION"
•200 PRINT:PRINT "{YL}     EEEEE
 EEEEE    EEEEE    EEEEE   "
•205 PRINT
•210 PRINT TAB(11);"{BL}YOUR CHOIC
E [1-4]?"
•220 GET CHOICE$:IF CHOICE$="" THE
N 220
•225 REM [***  PLAYER LEVEL  ***]
•226 REM ========================
•230 PRINT "{SC}":POKE 53280,0:POK
E 53281,4
•240 PRINT TAB(9);"{YL}{CD}EEEE PL
AYER LEVEL EEEE :PRINT TAB(14);"E
EEEEEEEEEEE"
•250 FOR A=1 TO 5:PRINT:NEXT A
•260 PRINT TAB(11);"{BK}[1]..AGES
6 TO 9":PRINT:PRINT TAB(11);"[2].
.AGES 10 TO 14"
•270 PRINT:PRINT TAB(11);"[3]..AGE
S 15 TO ADULT":FOR A=1 TO 6:PRINT
:NEXT A
•280 PRINT TAB(12);"{WH}YOUR CHOIC
E [1-3]?"
•290 GET LEV$:IF LEV$="" THEN 290
•300 LEV=VAL(LEV$):CHOICE=VAL(CHOI
CE$)
•310 IF LEV=1 THEN V=10:DIG=1:T=15
:GOTO 360
•320 IF LEV=2 THEN V=50:DIG=10:T=1
2:GOTO 360
•330 IF LEV=3 THEN V=100:DIG=25:T=
10:GOTO 360
•360 ON CHOICE GOTO 500,1000,1500,
2000
•497 REM
•498 REM [ ****  ADDITION  **** ]
•499 REM ========================
•500 PRINT "{SC}":POKE 53280,0:POK
E 53281,5
•510 PRINT TAB(11);"{BK}* A D D I
T I O N *":PRINT:PRINT TAB(17);"{
BL}LEVEL";LEV
•520 PROB=PROB+1:X=INT(RND(0)*V)+D
IG:Y=INT(RND(0)*V)+DIG
•530 PRINT TAB(15);"{CD}{BL}PROBLE
M:";PROB
•540 FOR A=1 TO 3:PRINT:NEXT A
•550 PRINT "{BK}       ACCCCCCCCCC
CCCCCCCCCCCCCCCS"
•560 PRINT "        B
            B"
•570 PRINT "        B
            B"
•580 PRINT "        B
            B"
•590 PRINT "       ZCCCCCCCCCCCCCC
CCCCCCCCCCCX"
•595 FOR D=1 TO 500:NEXT D
•600 FOR L=0 TO 24:POKE S+L,0:NEXT
•605 POKE S+1,33:POKE S,135:POKE S
+5,9:POKE S+24,10:POKE S+4,33
•610 PRINT TAB(11);"{WH}{CU}{CU}{C
U}";X;" + ";Y;" = "
•620 TI$="000000":B=0:Z=12
•630 B=B+1:FOR A=1 TO Z:PRINT:NEXT
•640 GET AN$(B):PRINT TAB(16);"{BK
}{CU}TIME:";"{PU}";INT(TI/60)
•645 IF INT(TI/60)>=T THEN 850
•650 IF AN$(B)="" THEN 640
•660 PRINT TAB(24+B);"{WH}{CU}{CU}
{CU}{CU}{CU}{CU}{CU}{CU}{CU}{CU}{
CU}{CU}{CU}";AN$(B):IF AN$(B)=CHR
$(13) THEN 670
•665 GOTO 630
•670 A$=AN$(1)+AN$(2)+AN$(3)
•675 ANS=VAL(A$):IF ANS<>X+Y THEN
780
•680 CENT=INT(40-LEN("CORRECT"+NAM
E$))/2
•690 PRINT TAB(CENT);"{YL}{CD}{CD}
{CD}CORRECT ";NAME$;","
•700 FOR A=10 TO 0 STEP -1:POKE 53
280,A:FOR D=1 TO 100:NEXT D:NEXT
A:C=C+1
•710 PRINT TAB(13);"{CD}YOU NOW HA
VE ";C
•720 PRINT TAB(13);"{CD}CORRECT AN
```

```
SWERS"
•730 FOR D=1 TO 2400:NEXT D
•740 IF PROB=10 THEN 3000
•750 GOTO 500
•760 REM [ * WRONG ANSWER * ]
•770 REM ===================
•780 CENT=INT(40-LEN("SORRY"+NAME$
))/2
•785 FOR L=0 TO 24:POKE S+L,0:NEXT
:POKE S+1,4:POKE S,48
•787 POKE S+5,10:POKE S+24,15:POKE
 S+4,33
•790 PRINT TAB(CENT);"{RD}{CD}{CD}
{CD}SORRY ";NAME$;","
•800 PRINT TAB(11);"{CD}BUT THE AN
SWER IS...":PRINT TAB(18);"{CD}";
X+Y
•810 FOR D=1 TO 2400:NEXT D
•820 IF PROB=10 THEN 3000
•830 GOTO 500
•840 REM [ *** TIME LIMIT *** ]
•841 REM ======================
•850 CENT=INT(40-LEN("SORRY"+NAME$
))/2
•855 FOR L=0 TO 24:POKE S+L,0:NEXT
:POKE S+1,4:POKE S,48
•857 POKE S+5,10:POKE S+24,15:POKE
 S+4,33
•860 PRINT TAB(CENT);"{RD}{CU}{CU}
{CU}{CU}{CU}{CU}{CU}{CU}{CU}SORRY
 ";NAME$;","
•870 PRINT TAB(11);"{CD}BUT THE AN
SWER IS...":PRINT TAB(18);"{CD}";
X+Y
•880 FOR D=1 TO 2400:NEXT D
•890 IF PROB=10 THEN 3000
•900 GOTO 500
•997 REM
•998 REM [ *** SUBTRACTION *** ]
•999 REM ======================
•1000 PRINT "{SC}":POKE 53280,0:PO
KE 53281,8
•1010 PRINT TAB(8);"{BK}* S U B T
R A C T I O N *":PRINT:PRINT TAB(
17);"{BL}LEVEL";LEV
•1020 PROB=PROB+1:X=INT(RND(0)*V)+
DIG+V:Y=INT(RND(0)*V)+1
•1030 PRINT TAB(15);"{CD}{BL}PROBL
EM:";PROB
•1040 FOR A=1 TO 3:PRINT:NEXT A
•1050 PRINT "{BK}          ACCCCCCCCC
CCCCCCCCCCCCCCCCS"
•1060 PRINT "          B
         B"
•1070 PRINT "          B
         B"
•1080 PRINT "          B
         B"
•1090 PRINT "          ZCCCCCCCCCCCCC
CCCCCCCCCCCCX"
•1095 FOR D=1 TO 500:NEXT D
•1100 FOR L=0 TO 24:POKE S+L,0:NEX
T
•1105 POKE S+1,33:POKE S,135:POKE
S+5,9:POKE S+24,10:POKE S+4,33
•1110 PRINT TAB(11);"{WH}{CU}{CU}{
CU}";X;" - ";Y;" = "
•1120 TI$="000000":B=0:Z=12
•1130 B=B+1:FOR A=1 TO Z:PRINT:NEX
T
•1140 GET AN$(B):PRINT TAB(16);"{B
K}{CU}TIME:";"{WH}";INT(TI/60)
•1145 IF INT(TI/60)>=T THEN 1350
•1150 IF AN$(B)="" THEN 1140
•1160 PRINT TAB(25+B);"{WH}{CU}{CU
}{CU}{CU}{CU}{CU}{CU}{CU}{CU}{CU}
{CU}{CU}{CU}";AN$(B):IF AN$(B)=CH
R$(13) THEN 1170
•1165 GOTO 1130
•1170 A$=AN$(1)+AN$(2)+AN$(3)
•1175 ANS=VAL(A$):IF ANS<>X-Y THEN
 1275
•1180 CENT=INT(40-LEN("CORRECT"+NA
ME$))/2
•1190 PRINT TAB(CENT);"{YL}{CD}{CD
}{CD}CORRECT ";NAME$;","
•1200 FOR A=10 TO 0 STEP -1:POKE 5
3280,A:FOR D=1 TO 120:NEXT D:NEXT
 A:C=C+1
•1210 PRINT TAB(13);"{CD}YOU NOW H
AVE ";C
•1220 PRINT TAB(13);"{CD}CORRECT A
NSWERS"
•1230 FOR D=1 TO 2400:NEXT D
•1240 IF PROB=10 THEN 3000
•1250 GOTO 1000
•1260 REM [ * WRONG ANSWER * ]
•1270 REM ===================
•1275 FOR L=0 TO 24:POKE S+L,0:NEX
T:POKE S+1,4:POKE S,48
•1277 POKE S+5,10:POKE S+24,15:POK
E S+4,33
•1280 CENT=INT(40-LEN("SORRY"+NAME
$))/2
•1290 PRINT TAB(CENT);"{LG}{CD}{CD
}{CD}SORRY ";NAME$;","
```

```
•1300 PRINT TAB(11);"{CD}BUT THE A
 NSWER IS...":PRINT TAB(18);"{CD}"
 ;X-Y
•1310 FOR D=1 TO 2400:NEXT D
•1320 IF PROB=10 THEN 3000
•1330 GOTO 1000
•1340 REM [ *** TIME LIMIT *** ]
•1341 REM ======================
•1350 CENT=INT(40-LEN("SORRY"+NAME
 $))/2
•1360 PRINT TAB(CENT);"{LG}{CU}{CU
 }{CU}{CU}{CU}{CU}{CU}{CU}{CU}SORR
 Y ";NAME$;","
•1370 PRINT TAB(11);"{CD}BUT THE A
 NSWER IS...":PRINT TAB(18);"{CD}"
 ;X-Y
•1380 FOR D=1 TO 2400:NEXT D
•1390 IF PROB=10 THEN 3000
•1400 GOTO 1000
•1500 PRINT "{SC}":POKE 53280,0:PO
 KE 53281,6
•1510 PRINT TAB(11);"{BK}* M U L T
  I P L Y *":PRINT:PRINT TAB(17);"
 {YL}LEVEL";LEV
•1520 PROB=PROB+1:X=INT(RND(0)*V)+
 DIG:Y=INT(RND(0)*V)+DIG
•1530 PRINT TAB(15);"{CD}{YL}PROBL
 EM:";PROB
•1540 FOR A=1 TO 3:PRINT:NEXT A
•1550 PRINT "{BK}          ACCCCCCCCC
 CCCCCCCCCCCCCCCCS"
•1560 PRINT "          B
          B"
•1570 PRINT "          B
          B"
•1580 PRINT "          B
          B"
•1590 PRINT "          ZCCCCCCCCCCCCC
 CCCCCCCCCCCCX"
•1595 FOR D=1 TO 500:NEXT D
 1600 FOR L=0 TO 24:POKE S+L,0:NEX
 T
•1605 POKE S+1,33:POKE S,135:POKE
 S+5,9:POKE S+24,10:POKE S+4,33
•1610 PRINT TAB(11);"{WH}{CU}{CU}{
 CU}";X;" X ";Y;" = "
•1620 TI$="000000":B=0:Z=12
•1630 B=B+1:FOR A=1 TO Z:PRINT:NEX
 T
•1640 GET AN$(B):PRINT TAB(16);"{B
 K}{CU}TIME:";"{PU}";INT(TI/60)
•1645 IF INT(TI/60)>=T THEN 1850
•1650 IF AN$(B)="" THEN 1640
•1660 PRINT TAB(24+B);"{WH}{CU}{CU
 }{CU}{CU}{CU}{CU}{CU}{CU}{CU}{CU}
 {CU}{CU}{CU}";AN$(B):IF AN$(B)=CH
 R$(13) THEN 1670
•1665 GOTO 1630
•1670 A$=AN$(1)+AN$(2)+AN$(3)
•1675 ANS=VAL(A$):IF ANS<>X*Y THEN
  1780
•1680 CENT=INT(40-LEN("CORRECT"+NA
 ME$))/2
•1690 PRINT TAB(CENT);"{YL}{CD}{CD
 }{CD}CORRECT ";NAME$;","
•1700 FOR A=10 TO 0 STEP -1:POKE 5
 3280,A:FOR D=1 TO 100:NEXT D:NEXT
  A:C=C+1
•1710 PRINT TAB(13);"{CD}YOU NOW H
 AVE ";C
•1720 PRINT TAB(13);"{CD}CORRECT A
 NSWERS"
•1730 FOR D=1 TO 2400:NEXT D
•1740 IF PROB=10 THEN 3000
•1750 GOTO 1500
•1760 REM [ * WRONG ANSWER * ]
•1770 REM ====================
•1780 CENT=INT(40-LEN("SORRY"+NAME
 $))/2
•1785 FOR L=0 TO 24:POKE S+L,0:NEX
 T:POKE S+1,4:POKE S,48
•1787 POKE S+5,10:POKE S+24,15:POK
 E S+4,33
•1790 PRINT TAB(CENT);"{RD}{CD}{CD
 }{CD}SORRY ";NAME$;","
•1800 PRINT TAB(11);"{CD}BUT THE A
 NSWER IS...":PRINT TAB(18);"{CD}"
 ;X*Y
•1810 FOR D=1 TO 2400:NEXT D
•1820 IF PROB=10 THEN 3000
•1830 GOTO 1500
•1840 REM [ *** TIME LIMIT *** ]
•1841 REM ======================
•1850 CENT=INT(40-LEN("SORRY"+NAME
 $))/2
•1855 FOR L=0 TO 24:POKE S+L,0:NEX
 T:POKE S+1,4:POKE S,48
•1857 POKE S+5,10:POKE S+24,15:POK
 E S+4,33
•1860 PRINT TAB(CENT);"{RD}{CU}{CU
 }{CU}{CU}{CU}{CU}{CU}{CU}{CU}SORR
 Y ";NAME$;","
•1870 PRINT TAB(11);"{CD}BUT THE A
 NSWER IS...":PRINT TAB(18);"{CD}"
 ;X*Y
•1880 FOR D=1 TO 2400:NEXT D
```

```
•1890 IF PROB=10 THEN 3000
•1900 GOTO 1500
•2000 PRINT "{SC}":POKE 53280,6:PO
KE 53281,9
•2010 PRINT TAB(11);"{BK}* D I V I
 S I O N *":PRINT:PRINT TAB(17);"
{YL}LEVEL";LEV
•2020 PROB=PROB+1:W=INT(RND(0)*V)+
DIG:Y=INT(RND(0)*10)+1
•2030 PRINT TAB(15);"{CD}{YL}PROBL
EM:";PROB
•2040 FOR A=1 TO 3:PRINT:NEXT A
•2050 PRINT "{BK}         ACCCCCCCCC
CCCCCCCCCCCCCCCCS"
•2060 PRINT "         B
          B"
•2070 PRINT "         B
          B"
•2080 PRINT "         B
          B"
•2090 PRINT "         ZCCCCCCCCCCCCC
•CCCCCCCCCCCCX"
•2095 FOR D=1 TO 500:NEXT D
•2100 FOR L=0 TO 24:POKE S+L,0:NEX
T
•2105 POKE S+1,33:POKE S,135:POKE
•S+5,9:POKE S+24,10:POKE S+4,33
•2110 X=INT(W*Y)
•2115 PRINT TAB(11);"{WH}{CU}{CU}{
CU}";X;" / ";Y;" = "
•2120 TI$="000000":B=0:Z=12
•2130 B=B+1:FOR A=1 TO Z:PRINT:NEX
T
•2140 GET AN$(B):PRINT TAB(16);"{B
K}{CU}TIME:";"{PU}";INT(TI/60)
•2145 IF INT(TI/60)>=T THEN 2350
•2150 IF AN$(B)="" THEN 2140
•2160 PRINT TAB(24+B);"{WH}{CU}{CU
}{CU}{CU}{CU}{CU}{CU}{CU}{CU}{CU}
{CU}{CU}{CU}";AN$(B):IF AN$(B)=CH
R$(13) THEN 2170
•2165 GOTO 2130
•2170 A$=AN$(1)+AN$(2)+AN$(3)
•2175 ANS=VAL(A$):IF ANS<>INT(X/Y)
 THEN 2280
•2180 CENT=INT(40-LEN("CORRECT"+NA
ME$))/2
•2190 PRINT TAB(CENT);"{YL}{CD}{CD
}{CD}CORRECT ";NAME$;","
•2200 FOR A=10 TO 0 STEP -1:POKE 5
•3280,A:FOR D=1 TO 100:NEXT D:NEXT
 A:C=C+1
•2210 PRINT TAB(13);"{CD}YOU NOW H
AVE ";C
•2220 PRINT TAB(13);"{CD}CORRECT A
NSWERS"
•2230 FOR D=1 TO 2400:NEXT D
•2240 IF PROB=10 THEN 3000
•2250 GOTO 2000
•2260 REM [ * WRONG ANSWER * ]
•2270 REM =====================
•2280 CENT=INT(40-LEN("SORRY"+NAME
$))/2
•2285 FOR L=0 TO 24:POKE S+L,0:NEX
T:POKE S+1,4:POKE S,48
•2287 POKE S+5,10:POKE S+24,15:POK
E S+4,33
•2290 PRINT TAB(CENT);"{GN}{CD}{CD
}{CD}SORRY ";NAME$;","
•2300 PRINT TAB(11);"{CD}BUT THE A
NSWER IS...":PRINT TAB(18);"{CD}"
;X/Y
•2310 FOR D=1 TO 2400:NEXT D
•2320 IF PROB=10 THEN 3000
•2330 GOTO 2000
•2340 REM [ *** TIME LIMIT *** ]
•2345 REM ======================
•2350 CENT=INT(40-LEN("SORRY"+NAME
$))/2
•2355 FOR L=0 TO 24:POKE S+L,0:NEX
T:POKE S+1,4:POKE S,48
•2357 POKE S+5,10:POKE S+24,15:POK
E S+4,33
•2360 PRINT TAB(CENT);"{GN}{CU}{CU
}{CU}{CU}{CU}{CU}{CU}{CU}{CU}SORR
Y ";NAME$;","
•2370 PRINT TAB(11);"{CD}BUT THE A
NSWER IS...":PRINT TAB(18);"{CD}"
;X/Y
•2380 FOR D=1 TO 2400:NEXT D
•2390 IF PROB=10 THEN 3000
•2400 GOTO 2000
•3000 REM *** FINAL RESULTS ***
•3010 REM ======================
•3020 PRINT "{SC}":POKE 53280,2:PO
KE 53281,15
•3030 PRINT TAB(8);"{RD}EEEEE{BL}
R E S U L T S {RD}EEEEE"
•3035 PRINT TAB(12);"&EEEEEEEEEEEE
EEE&"
•3040 PRINT TAB(10);"{BL}{CD}{CD}Y
OU SCORED ";C;"OUT OF"
•3050 PRINT TAB(15);"{CD}10 CORREC
T"
•3060 P=INT(C*100)/10
•3070 PRINT TAB(11);"{CD}FOR A MAR
```

```
K OF ";P;"%"
•3080 PRINT "{RD}{CD}{CD}     EEEEE
     EEEEE    EEEEE    EEEEE"
•3085 FOR D=1 TO 2000:NEXT D
•3090 IF P<=60 THEN 3120
•3100 IF P<=90 THEN 3150
•3110 IF P>90 THEN 3180
•3120 CENT=INT(40-LEN("COME ON"+NA
ME$))/2
•3130 PRINT TAB(CENT);"{BL}{CD}{CD
}COME ON ";NAME$;",":PRINT TAB(12
);"{CD}I KNOW YOU CAN DO"
•3140 PRINT TAB(12);"{CD}BETTER TH
AN THAT":GOTO 3220
•3150 CENT=INT(40-LEN("NOT BAD"+NA
ME$))/2
•3160 PRINT TAB(CENT);"{BL}{CD}{CD
}NOT BAD ";NAME$;","
•3170 PRINT TAB(11);"{CD}I THINK Y
OU SHOULD":PRINT TAB(11);"{CD}TRY
 A LITTLE HARDER"
•3175 GOTO 3220
•3180 CENT=INT(40-LEN("CONGRATULAT
IONS"+NAME$))/2:FOR A=1 TO 3:PRIN
T:NEXT A
•3185 FOR A=1 TO 7:PRINT TAB(CENT)
;"{RD}{RV}{CU}CONGRATULATIONS ";N
AME$;","
•3190 FORD=1 TO 200:NEXT D:PRINT T
AB(CENT);"{CU}CONGRATULATIONS ";N
AME$;","
•3195 FOR D=1 TO 200:NEXT D:NEXT A
•3200 PRINT TAB(12);"{BL}{CD}YOU G
OT A PERFECT":PRINT TAB(16);"{CD}
SCORE !!!"
•3220 PRINT TAB(12);"{RD}{CD}{CD}{
CD}TRY AGAIN [Y/N]?";
•3230 GET X$:IF X$="" THEN 3230
•3240 IF X$<>"Y" THEN END
•3250 GOTO 90
```

## BUG REPELLENT LINE CODES FOR MATH MASTER

```
LINE # 10:FI
LINE # 15:MI
LINE # 20:FI
LINE # 90:NA
LINE # 100:DA
LINE # 110:MI
LINE # 120:LI
LINE # 130:EA
LINE # 140:IE
LINE # 150:EI
LINE # 160:KI
LINE # 161:CA
LINE # 162:FE
LINE # 165:FI
LINE # 166:KE
LINE # 170:II
LINE # 180:II
LINE # 190:EI
LINE # 200:EI
LINE # 205:GE
LINE # 210:JA
LINE # 220:NA
LINE # 225:CE
LINE # 226:AM
LINE # 230:EA
LINE # 240:EI
LINE # 250:NE
LINE # 260:BI
LINE # 270:FE
LINE # 280:HA
LINE # 290:AI
LINE # 300:DE
LINE # 310:MA
LINE # 320:EA
LINE # 330:EA
LINE # 360:IA
LINE # 497:DM
LINE # 498:CE
LINE # 499:AM
LINE # 500:EE
LINE # 510:CA
LINE # 520:CE
LINE # 530:GA
LINE # 540:NE
LINE # 550:MI
LINE # 560:DI
LINE # 570:DI
LINE # 580:DI
LINE # 590:EA
LINE # 595:EA
LINE # 600:FI
LINE # 605:KE
LINE # 610:LA
LINE # 620:DA
LINE # 630:CI
LINE # 640:GE
LINE # 645:MI
LINE # 650:CA
LINE # 660:HI
LINE # 665:NI
LINE # 670:DM
LINE # 675:BA
LINE # 680:GA
LINE # 690:OI
LINE # 700:EE
LINE # 710:KE
LINE # 720:CA
LINE # 730:MA
LINE # 740:GA
LINE # 750:LA
LINE # 760:ME
LINE # 770:AM
LINE # 780:GA
LINE # 785:GA
LINE # 787:KE
LINE # 790:OI
LINE # 800:JE
LINE # 810:MA
LINE # 820:GA
LINE # 830:LA
LINE # 840:CE
LINE # 841:AM
LINE # 850:GA
LINE # 855:GA
LINE # 857:KE
LINE # 860:OI
LINE # 870:JE
LINE # 880:MA
LINE # 890:GA
LINE # 900:LA
LINE # 997:DM
LINE # 998:KE
LINE # 999:AM
LINE # 1000:FA
LINE # 1010:CA
LINE # 1020:OE
LINE # 1030:GA
LINE # 1040:NE
LINE # 1050:MI
LINE # 1060:DI
LINE # 1070:DI
LINE # 1080:DI
LINE # 1090:EA
LINE # 1095:EA
LINE # 1100:FI
LINE # 1105:KE
LINE # 1110:LA
LINE # 1120:DA
LINE # 1130:CI
LINE # 1140:GE
LINE # 1145:HI
LINE # 1150:FA
LINE # 1160:OI
LINE # 1165:MI
LINE # 1170:DM
LINE # 1175:AM
LINE # 1180:GA
LINE # 1190:OI
LINE # 1200:EE
LINE # 1210:KE
LINE # 1220:CA
LINE # 1230:MA
LINE # 1240:GA
LINE # 1250:KA
```

```
LINE # 1260:ME
LINE # 1270:AM
LINE # 1275:GA
LINE # 1277:KE
LINE # 1280:GA
LINE # 1290:OI
LINE # 1300:JM
LINE # 1310:MA
LINE # 1320:GA
LINE # 1330:KA
LINE # 1340:CE
LINE # 1341:AM
LINE # 1350:GA
LINE # 1360:OI
LINE # 1370:JM
LINE # 1380:MA
LINE # 1390:GA
LINE # 1400:KA
LINE # 1500:EI
LINE # 1510:CA
LINE # 1520:CE
LINE # 1530:GA
LINE # 1540:NE
LINE # 1550:MI
LINE # 1560:DI
LINE # 1570:DI
LINE # 1580:DI
LINE # 1590:EA
LINE # 1595:EA
LINE # 1600:FI
LINE # 1605:KE
LINE # 1610:LA
LINE # 1620:DA
LINE # 1630:CI
LINE # 1640:GE
LINE # 1645:MI
LINE # 1650:KA
LINE # 1660:DI
LINE # 1665:BI
LINE # 1670:DM
LINE # 1675:FA
LINE # 1680:GA
LINE # 1690:OI
LINE # 1700:EE
LINE # 1710:KE
LINE # 1720:CA
LINE # 1730:MA
LINE # 1740:GA
LINE # 1750:PA
LINE # 1760:ME
LINE # 1770:AM
LINE # 1780:GA
LINE # 1785:GA
LINE # 1787:KE
LINE # 1790:OI
LINE # 1800:KE
LINE # 1810:MA
LINE # 1820:GA
LINE # 1830:PA
LINE # 1840:CE
LINE # 1841:AM
LINE # 1850:GA
LINE # 1855:GA
LINE # 1857:KE
LINE # 1860:OI
LINE # 1870:KE
LINE # 1880:MA
LINE # 1890:GA
LINE # 1900:PA
LINE # 2000:FE
LINE # 2010:CA
LINE # 2020:OE
LINE # 2030:GA
LINE # 2040:NE
LINE # 2050:MI
LINE # 2060:DI
LINE # 2070:DI
LINE # 2080:DI
LINE # 2090:EA
LINE # 2095:EA
LINE # 2100:FI
LINE # 2105:KE
LINE # 2110:IM
LINE # 2115:LA
LINE # 2120:DA
LINE # 2130:CI
LINE # 2140:GE
LINE # 2145:JI
LINE # 2150:HA
LINE # 2160:AI
LINE # 2165:OI
LINE # 2170:DM
LINE # 2175:CA
LINE # 2180:GA
LINE # 2190:OI
LINE # 2200:EE
LINE # 2210:KE
LINE # 2220:CA
LINE # 2230:MA
LINE # 2240:GA
LINE # 2250:MA
LINE # 2260:ME
LINE # 2270:AM
LINE # 2280:GA
LINE # 2285:GA
LINE # 2287:KE
LINE # 2290:OI
LINE # 2300:KM
LINE # 2310:MA
LINE # 2320:GA
LINE # 2330:MA
LINE # 2340:CE
LINE # 2345:AM
LINE # 2350:GA
LINE # 2355:GA
LINE # 2357:KE
LINE # 2360:OI
LINE # 2370:KM
LINE # 2380:MA
LINE # 2390:GA
LINE # 2400:MA
LINE # 3000:PI
LINE # 3010:AM
LINE # 3020:DM
LINE # 3030:EI
LINE # 3035:DI
LINE # 3040:OI
LINE # 3050:HI
LINE # 3060:DI
LINE # 3070:LA
LINE # 3080:EI
LINE # 3085:MA
LINE # 3090:IA
LINE # 3100:JI
LINE # 3110:LA
LINE # 3120:GA
LINE # 3130:MA
LINE # 3140:BA
LINE # 3150:GA
LINE # 3160:OI
LINE # 3170:OI
LINE # 3175:BA
LINE # 3180:NE
LINE # 3185:OI
LINE # 3190:OI
LINE # 3195:LE
LINE # 3200:AA
LINE # 3220:IM
LINE # 3230:JI
LINE # 3240:HI
LINE # 3250:BI
LINES: 275
```

# COMMODARES

**From Page 40**

**Problem 4-1**

```
•0 REM COMMODARES #5
•1 REM << PROBLEM 4-1 : SQUARED SU
M >>
•2 REM - FIND NUMBERS EQUAL TO THE
 SQUARE OF THE SUM OF THEIR DIGIT
S
•5 FOR J=1 TO 100
•10 SUM=0 : REM INITIAL SUM OF DIG
ITS
•15 JJ=J*J : REM JJ IS A SQUARE
•20 J$=STR$(JJ) :REM CONVERT JJ TO
 A STRING VARIABLE J$
•25 REM -- LINES 30 - 60 TAKE EACH
 DIGIT OF J$ AND ADD ITS VALUE TO
 SUM ---
•30 FOR D=2 TO LEN(J$)
•40 V=VAL(MID$(J$,D,1))
•50 SUM=SUM+V
•60 NEXT D
•65 SS=SUM*SUM : REM > GET THE SQU
ARE OF THE SUM OF THE DIGITS
•70 IF ABS(SS-JJ)<1E-4 THEN PRINT
JJ"IS A SOLUTION"
•80 NEXT J
```

### Problem 4-2

```
•0 REM COMMODARES #5
•1 REM <<< PROBLEM 4-2 : SUM OF CU
 BES >>>
•2 REM - FIND NUMBERS EQUAL TO THE
  SUM OF THE CUBES OF THEIR DIGITS
•10 FOR J=1 TO 1000
•15 J$=STR$(J):SUM=0
•20 FOR D=2 TO LEN(J$)
•40 V=VAL(MID$(J$,D,1))
•50 SUM=SUM+V*V*V
•60 NEXT D
•70 IF ABS(J-SUM)<1E-4 THEN PRINT
 J"IS A SOLUTION"
•80 NEXT J
```

### Problem 4-4

```
•0 REM COMMODARES #5
•1 REM << PROB 4-4 : RANDOM REPETI
 TION
•2 REM RVS ON=CHR$(18):RVS OFF=CHR
 $(146)
•3 REM - FLAG IS TRUE (-1) WHEN PR
 EVIOUS  A% EQUALS PREVIOUS B%
•5 FLAG=0
•10 A%=RND(0)*10
•20 B%=RND(0)*10
•30 IF NOT FLAG THEN PRINT CHR$(14
 6);
•35 REM - LINES 40 & 50 DETERMINE
 THE NEXT STATE OF THE FLAG
•40 FLAG=0
•50 IF A%=B% THEN PRINT CHR$(18);:
 FLAG=-1
•60 PRINT A%;
•70 A%=B%
•80 GOTO 20
```

## BUG REPELLENT LINE CODES FOR COMMODARES

### Problem 4-1

```
LINE # 0:LM          LINE # 30:CE
LINE # 1:BI          LINE # 40:HM
LINE # 2:JM          LINE # 50:NI
LINE # 5:BA          LINE # 60:CA
LINE # 10:JM         LINE # 65:JM
LINE # 15:BE         LINE # 70:II
LINE # 20:LA         LINE # 80:DI
LINE # 25:EM         LINES: 15
```

### Problem 4-2

```
LINE # 0:LM          LINE # 1:GI
LINE # 2:JM          LINE # 50:BI
LINE # 10:OA         LINE # 60:CA
LINE # 15:IA         LINE # 70:II
LINE # 20:CE         LINE # 80:DI
LINE # 40:HM         LINES: 11
```

### Problem 4-4

```
LINE # 0:LM          LINE # 35:OE
LINE # 1:AA          LINE # 40:OA
LINE # 2:LE          LINE # 50:FM
LINE # 3:LE          LINE # 60:EE
LINE # 5:OA          LINE # 70:KE
LINE # 10:CI         LINE # 80:OA
LINE # 20:CI         LINES: 14
LINE # 30:FE
```

# THAT'S MY PROGRAM

**From page 62**

gram, this makes a considerable difference.

We hope you now have some insight as to what goes on "behind the scenes" as you type in your programs.

| MEMORY LOCATION | CONTENTS | COMMENTS |
|---|---|---|
| 2048 | 0 | First byte is always zero |
| | | LINE 10 |
| 2049 | 13 | Link-points to 2061 |
| 2050 | 8 | |
| 2051 | 10 | Line number = 10 |
| 2052 | 0 | |
| 2053 | 153 | Token for print |
| 2054 | 34 | ASC for "DEMO" |
| 2055 | 68 | |
| 2056 | 69 | |
| 2057 | 77 | |
| 2058 | 79 | |
| 2059 | 34 | |
| 2060 | 0 | End of line 10 |
| | | LINE 20 |
| 2061 | 35 | Link-points to 2083 |
| 2062 | 8 | |
| 2063 | 20 | Line number = 20 |
| 2064 | 0 | |
| 2065 | 129 | Token for "FOR" statement |
| 2066 | 76 | ASC for "LOOP" |
| 2067 | 79 | |
| 2068 | 79 | |

```
2069     80─┘
2070    178     Token for "="
2071     49     ASC for "1"
2072    164     Token for "TO"
2073     57─┐
2074     57 ├ASC for "999"
2075     57─┘
2076     58     Colon
2077    130     Token for "NEXT"
2078     76─┐
2079     79 │
            ├ASC for "LOOP"
2080     79 │
2081     80─┘
2082      0     End of line 20

            LINE 30
2083     41─┐
            ├Link-points to 2078
2084      8─┘
2085     30─┐
            ├Line number = 30
2086      0─┘
2087    128     Token for "END"
2088      0     End of line
2089      0─┐
            ├0 link means end of
2090      0─┘  program
```

# SPREAD-SHEETS

*Continued from page 34*

ing not-so-standard formulas. One calculates trends using the method of least square. If you have a list of values 4,7,12,15, *Omnicalc* will tell you the most likely number to follow. There is a built-in formula to amortize interest or depreciation. *Omnicalc* comes with *Plot 64* that will print bar charts, pie charts and line graphs to the screen or a Gemini 10x, Epmson mx80 or Commodore 1525.

(Note: *Omnicalc* is currently being phased out in favor of *Multiplan*. Once the present inventory is sold, no more will be made. Hes will send a copy of *Multiplan* to any *Omnicalc* owner who mails in his disk plus $30.00)

### *PRACTICALC (FROM COMPUTER SOFTWARE ASSOCIATES.)*

A spreadsheet with up to 250 rows or 100 columns can be created as long as you do not exceed 2000 cells. Columns can be displayed from 3 to 30 characters in width. Mathematical operations include addition, subtraction, division, multiplication, exponentials, inequalities, sine, cosine, and tangent. A list can be alphabetically or numerically sorted. The maximum, minimum, and average value can be automatically found from the list. The program can display bar charts in a high and low resolution mode, print low resolution graphs to most printers, and print high resolution graphs to the 1525 printer.

### *SPREADSHEET ASSISTANT (FROM RAINBOW CORPORATION)*

This program can be configured to create a sheet with up to 200 rows or 52 columns within the constraints of memory. Column widths can be varied from 1 to 20 characters. Calculations are limited to addition, subtraction, multiplication, and division. The program is rather limited in scope, but the documentation is excellent and the program easy to use.

---

On pages 94 and 95 you'll find a chart comparing the capabilities of the spreadsheets reviewed. Following is an explanation of the chart categories.

Relative Copy: Enables you to copy a row or column automatically and make any changes in formulas that reference the same column.

IF THEN Command: You can develop formulas using the BASIC IF THEN commands, e.g. If A1 > $100 THEN 5 ELSE 10: This formula entered into a cell would place the number 5 in the cell referenced if the value in A1 were greater than $100; otherwise it would place the number 10 in the cell.

R/C Insert: You can insert an additional blank row column in the spreadsheet you are creating even after you have already entered data.

Rejustify: Names, values and formulas are generally aligned to the left or right of the cell so that they appear even throughout the spreadsheet. A rejustify feature allows you to individually change the justification to align the entry to the opposite side of the cell.

Split Screen: The ability to view two sections of a spreadsheet simultaneously. The screen is divided into two or more sections. You can choose which part of the spreadsheet you wish to view in each section independent of the other sections.

Adj. Col Width: Generally a spreadsheet automatically sets the width of a column displayed. It it has

## SPREADSHEET FEATURES

| | Relative Copy | If Then Command | R/C Insert | Rejustify | Split Screen | Adj. Col Width | Logic Operators | Fix Titles | Graph | Sort | Search | Link |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BCalc | – | – | – | – | – | – | – | – | – | – | – | – |
| Busicalc | + | – | + | + | – | + | – | – | – | – | – | – |
| Calc Result Adv. | + | + | + | + | + | + | + | + | + | – | + | – |
| Calc Result Easy | + | + | + | + | – | + | + | + | + | – | – | – |
| ESP>Calc | – | – | + | – | – | – | – | – | – | – | – | – |
| Multiplan | + | + | + | + | + | + | + | + | – | + | – | + |
| Omnicalc | + | + | + | + | – | + | + | + | + | – | – | + |
| Practicalc | + | – | + | + | – | + | + | + | + | + | + | – |
| Spread. Assistant | + | – | + | + | – | + | – | – | – | – | – | – |

an adjustable column width you can change the width to better suit your needs.

Logic Operators: You can create formulas using the logical operators AND, OR, or NOT.

Fix Titles: Allows you to hold a column or row in view while moving the rest of the spreadsheet.

Graph: Some sort of graph relating to a column or row of values can be displayed on the screen. Be careful if you are looking for a program that will print it out.

Sort: A list of names or numbers can be automatically sorted in alphabetical or numerical order.

Search: You can enter a command and a name or number and the program will search through the entire spreadsheet looking for an identical match.

Link: Perhaps one of the most powerful features of a spreadsheet. Allows you to create formulas that can reference other spreadsheets. When calculations are made, the program will look up the value on the other spreadsheet and insert it in the new one.

# RUPERT REPORT

*Continued from page 70*

Type the following program to see just how arrays are stored:

```
5 N=3
10 DIM A1%(N)
20 FOR I=0 TO N:A1%(I)=I+9:NEXT
30 MEM=0:M1=0:M2=0
40 M1=PEEK(47)+256*PEEK(48)
50 M2=PEEK(49)+256*PEEK(50)
60 FOR MEM=M1 TO M2-1
70 PRINT PEEK(MEM);:NEXT
```

We are using an integer array, since integer values are easily recognizable in memory (unlike floating point values as we saw last month). Line 30 is important to ensure that all variables have been defined before the pointers are read. The reason is that as new variables are defined, the entire array storage area is moved, and consequently the pointers are changed.

When this program is run, the result is:

| Max Col. Width | Max Cols. | Max Cells | Max Rows | Price |
|---|---|---|---|---|
| 7 | 99 | 3500 | 35 | $44.95 |
| 18 | 99 | 1100 | 99 | 69.00 |
| 18 | 63 | 1500 | 63 | 149.95 |
| 18 | 63 | 1500 | 63 | 79.95 |
| 8 | 645 | 2000 | 645 | 47.50 |
| | 63 | M | 255 | 99.00 |
| 20 | 65 | M | 255 | 49.00 |
| 38 | 100 | 2000 | 250 | 35.00 |
| 20 | 52 | M | 200 | 125.00 |

```
193 177 15 0 1 0 4 0 9 0 10 0 11
0 12
```

The variable name A1% is cleverly disguised as "193 177". This is "65 49" after subtracting 128 from each. And "65 49" is the ASCII representation for "A1" (see Appendix C of the *Programmer's Reference Guide)*. The added 128 tells the BASIC interpreter that this is an integer variable.

The 15 after the variable name is simply a count of the total number of memory locations used by this array. The next two values (bytes) indicate that this array has one dimension. The "0 4" tells the number of elements in that dimension. A1% has elements 0, 1, 2, and 3. The remaining pairs of bytes are the values of the elements: 9, 10, 11, and 12, assigned in line 20 of the program.

Things start getting complicated when arrays of two or more dimensions (matrices) are used. It should not be difficult, however, to modify this program and learn more about them.

We only made it through one-dimensional arrays this month. In a future column we will investigate uses for arrays of more than one dimension. In the meantime, don't let your computer catch you dealing from the bottom of the deck.

# READER SERVICE INDEX

**Access.** Retrieval of information from computer memory.

**Access time.** Time needed to retrieve information from computer memory.

**Address.** Location (identified by a number) of a byte of information in the computer's memory.

**Alphanumeric.** Alphabetic and numeric characters.

**Analog signal.** A continuous electronic signal of any frequency or strength that represents a condition (i.e.: the position of the game control paddles).

**Applications software.** Programs that will instruct the computer to perform either a specific task or a series of related tasks, usually relating to business or home uses.

**ASCII.** The American Standard Code for Information Interchange. This code allows two computers, which may entertain different languages, to communicate.

**Assembler.** A program that converts information in the computer's memory into a binary code for proper execution.

**Assembly language.** Machine language that has been translated into mnemonic codes, thus making it easier for the programmers to remember. A three letter code would be the equivalent to a string of eight digits.

**BASIC.** Beginner's All-purpose Symbolic Instruction Code. Easy to use, popular programming language that is widely used with microcomputers.

**Baud.** A unit (bits per second) that measures the rate of speed at which data is translated from one device to another. (Heck, every baud *we've* ever known has been fast.)

**Binary code.** A system of numbers that uses two digits, 0 and 1, to express all characters (both numbers and letters). The computer then uses this string of numbers to process information.

**Bit.** An acronym for *B*inary Dig*IT*. Represents either "0" or "1" in the binary code. Approximately eight bits (one byte) are required to represent one character.

**Bidirectional printing.** Special feature on some computer printers that allows the printer to print first right to left and then drop to the next line and print left to right. This feature speeds up the printing process.

**Board.** A thin, rectangular, flat electronic component that contains one or more layers of printed circuits. Chips and other electronic parts are often attached to a circuit board.

**Boot.** To start or restart a computer by transferring instructions ("Heel!"?) from a storage device, such as a tape or disk, into the computer's memory.

**Bug.** A logic error in a computer program that causes the program to dysfunction.

**Bus.** A conductor that allows data to be passed between the various computer components. Buses are manufactured that will allow different brands of components to be used with the same computer. (They can often be seen entering terminals.)

**Byte.** Plural of bit. There are eight bits in one byte. Each byte represents one character.

**CAI.** Computer Assisted Instruction. This refers to a variety of instructional and/or educational software.

**Card.** A circuit board which is attached to a standing computer, often to boost its memory capability.

**Cartridge.** Sometimes referred to as a ROM module or solid state cartridge. Device that contains a prerecorded program (such as a game).

**Cassette Tape Recorder.** Device used either to store data or to house prerecorded programs. Often, but not always, the same type used in audio recording. If so, a special interface may still be required. A much slower and less reliable medium than disk drive.

**C-BASIC.** Very popular non-interactive language that is utilized by 8080, 8085 and Z80 microprocessor computers. C-BASIC executes data at a more accelerated rate than the standard interpreter BASIC.

**Character.** A single symbol, letter, or number.

**Character set.** Total catalogue of alphanumeric, special, and punctuation characters accessible to a printer or terminal.

**Chip.** An integrated circuit, a quarter to three quarters of an inch square, that is etched on a tiny piece of silicon. Chips are the building blocks of computers and are able to contain anywhere from a few dozen to several thousand transistors or circuit elements. They perform many functions: calculations, memory operation and storage or controlling other chips. Up to 32,000 bytes of information may be stored in a single chip.

**CPU.** Central Processing Unit, or microprocessor. The electronic "core" of a computer. All information is passed through the CPU and all functions are executed from the CPU.

**Circuit Board.** see board.

**COBOL.** Acronym for Common Business Oriented Language. High level language for business applications.

**Command.** An instruction (e.g.: run, load) that tells the computer to do something.

**Compatibility.** The ability of the computer and any or all of its peripherals to be able to function in conjunction with one another.

**Compiler.** A program that translates any high level language into the binary code that is required for the computer.

**Computer.** A programmable electronic device, consisting of a CPU, memory and input/output (I/O) capability, that stores, retrieves, and processes data.

**Computer system.** A computer setup that consists of a computer terminal and its software and various peripherals.

**CP/M.** Control program for microcomputers. A single-user operating system for microcomputers that is in very wide use, especially for business applications.

**CRT.** Cathode Ray Tube. The most popular type of display screen. Often it is simply the screen of an ordinary television set though it can be a monitor specially designed for use with computers.

**Cursor.** Symbol, most often a small flashing square, that indicates where the next character will appear.

**Data.** All information that is processed by the computer.

**Datum.** Singular of data.

**Data Bank.** The central storage locations for all of the information that the computer has access to.

**Data base manager.** Program that enables user to enter files, organize matter and sort and retrieve information.

**Daisy wheel printer.** A letter quality impact printer that utilizes a daisy wheel, a circular printing element containing as many as a hundred characters.

**Descenders.** The lower case characters g,j,p,q, and y, which have tails that extend past the baseline formed by the other characters.

**Disk.** A flat, round device for storing computer data. Like a phonograph disc, it rotates and contains information in tracks. Unlike a phonograph disc, it is never removed from its protective jacket. A disk can be made of hard plastic (hard disk) or soft plastic (floppy disk, diskette).

**Diskette.** See disk.

**Disk drive.** A machine that rotates a disk for the purpose of entering new information or copying existent information from the disk into the com-

puter's memory.

**Display.** Generally a television-style screen, it shows the user the information contained in the terminal memory.

**Documentation.** Written instructions for operating hardware or software.

**DOS.** Disk Operating System. (See Operating System.)

**Dot matrix.** The method by which most printers form alphanumeric characters or graphics, by printing a series of dots.

**Double-density.** Refers to the process of doubling the amount of information that can be placed on a disk or cassette.

**Downtime.** When a computer is "down," or unoperational.

**Dual Intensity.** Refers to a printer that can produce bold type in additional to ordinary type.

**Editor.** A program which makes it possible to enter text into a computer.

**Electronic mail.** Telephone transmission (via a modem) of information between computer users.

**EPROM.** Erasable Programmable Read Only Memory. A device such as a cartridge or disk that can have data added to it or erased.

**Floppy disk.** See disk.

**FORTRAN.** Formula Translator. Advanced programming language used for complex mathematical operations.

**Emulator.** A program-translating device that enables software designed for a particular computer to run on a different one.

**Firmware.** ROM-contained data, built into the machine or added via software, that cannot be changed.

**Flow chart.** Diagram outlining procedure for writing a program.

**Format. (or Initialize)** To get a disk ready to accept information.

**Function key.** A key on a computer that can be assigned to perform a particular function.

**Full duplex.** Communication mode capable of simultaneous transmission and reception of data.

**Graphics.** Non-textual CRT displays: charts, symbols, etc.

**Graphics tablet.** A device which will transfer whatever is drawn on it onto the computer monitor.

**Half duplex.** Communication mode capable of transmission of data and reception of data, but not at the same time.

**Hard copy.** Information, when it has been transferred from memory to a computer printout, is hard copy.

**Hardware.** Mechanical, non-software components of a computer system: computer, disk drive, printer, etc.; also, the components of the components, such as transistors, circuits, etc. Also the components of the components of the . . .just kidding.

**Hard disk.** Nonremovable storage device that is faster and has a far greater storage capacity than a floppy disk. It is generally hermetically sealed within the unit.

**Hexadecimal.** A base-16 numeration method widely used with computers. Numbers run from 0 to 9, then from A to F. Hexadecimal (or "hex") numbers are identified with the suffix H.

**High-level language.** An easily learned programming language that resembles human languages such as English. Examples. BASIC. COBOL, FORTRAN. Low level languages (such as Assembly) require that every command and function be separately programmed.

**High resolution.** High-quality graphics capability when applied to a video terminal. An individual software program may itself contain high-resolution graphics, meaning detailed or colorful graphics. A subjective term and therefore frequently abused (especially on New Years Eve).

**Impact printer.** A typewriter-style printer that presses characters through a ribbon.

**Index hole.** A hole punched in a floppy disk that is used by the disk drive to locate the beginning of sector 0 on the disk.

**Instruction.** A command contained in a program.

**Interface.** A connecting device which can be electronic or can be contained in software, for making a computer compatible with a peripheral device.

**I/O.** Input/Output. The system of pathways which channel information into a computer (usually through a keyboard, joystick, mouse, light pen) and out of the computer (usually onto a screen or printer).

**K.** See "Kilobyte"

**Kilobyte.** See "K". Seriously, a kilobyte (or 1K) is equal to 1,024 bytes (often rounded off to 1,000 bytes), and is the measure of the memory hardware or software can contain (8K, 64K, etc., etc.). Obviously, the higher the number the more capable the hardware/software.

**Letter-quality.** Refers to a printer that uses formed characters, such as those contained on a daisy wheel. Compare dot matrix printer.

**Library.** Collection of computer programs.

**Light Pen.** A device that allows a user to draw line figures or make menu selections directly on the screen. Proper hardware port and software required.

**Load.** Entry of information into the computer from an external storage, such as cassette player or disk drive.

**LOGO.** Programming language useful for graphics. Primarily employed in young children's education, each command is built logically on the previous command.

**Loop.** Programming statement used to order repetition of a task. A particularly useful one is called a "fruitful loop."

**Machine language.** Language used by the computer, comprised of binary numbers, into which the computer must translate programming languages.

**Megabyte.** Equal to 1,048,576 bytes (1048 kilobytes). Often abbreviated Mb.

**Megahertz.** Electrical frequency measure equalling one million cycles per second. Often abbreviated MHz.

**Membrane.** A flat computer keyboard that has touch-sensitive areas for each character rather than full-travel or stepped keys.

**Memory.** The word used to indicate any given computer's capacity to store information (10K memory etc.). There are different kinds of memory (ROM, RAM) and, within any computer, different memory locations (for sound, screen displays etc.).

**Menu.** A screen display of programs, on a particular disk or options in a program.

**Microcomputer.** An integrated small computer. It contains a microprocessor, memory, and interfaces for inputting and outputting information. Perfect example? Commodore 64.

**Microprocessor.** The brains of the computer, where mathematics and logical functions are performed. Also called the CPU.

**Modem.** Modulator/Demodulator. A device that changes information in analog form into digital form, and vice versa, for the purpose of transmitting computer information across a telephone line.

**Monitor.** Screen for displaying computer information.

**Mouse.** An input device, usually containing a selection button. The user slides the mouse on the

desktop beside the computer, and the cursor will duplicate the mouse's movement on the screen. Used to make menu selections and re-arrange information.

**MP/M.** Multiprogramming control Program for Microprocessors. A variety of the CP/M operating system that can be used by several users at a time.

**Multi-user system.** A system with a central pool of data or applications that can be accessed simultaneously be several users.

**Number crunching.** Refers to a computer's carrying out intricate or dense arithmetic or numerical functions.

**Nybble.** Half a byte, or four bits. (Or two shaves and two haircuts.)

**Object code.** A code in binary form, produced by an assembler or compiler program. A source code, by contrast, must be translated by an assembler or compiler before it can be executed by the CPU.

**Octal.** A base-8 numeration method often used with microcomputers. Numbers run from 0 to 7, and are identified with the suffix Q.

**On-line.** The state of being hooked up to an active computer, as in the case of a printer or disk drive that is connected and operational.

**Operating system.** The program(s) that supplies the computer system with its operating information, including start-up instructions at the time the system is turned on and supervisory instructions each time a new program is loaded.

**Output.** The path system that channels information out of a computer (usually to a screen or printer).

**Overstriking.** A method of producing boldface type by directing the printer to hit a character more than once.

**Parallel.** An input/output system that submits 8 bits of data at a time. An interface would have to be installed between a computer's serial port and a parallel printer, for example.

**PASCAL.** A more sophisticated programming language than BASIC, using less memory and producing faster programs. Named after Blaise Pascal.

**PEEK.** A programming command generally meaning: examine (specified location) and report the value that is represented there.

**Peripheral.** A hardware accessory to a computer, such as a printer or a modem.

**Pixel.** Picture element. A dot of light on a TV or computer screen, the smallest light fragment that the computer can address. Graphics with high resolution are generally composed of very small, and therefore numerous pixels.

**Plotter.** Machine for printing lines or graphs.

**POKE.** A programming command that is used to place a new value into a specified memory location.

**Printer.** Machine used to print computer information onto paper. See dot matrix, letter quality, daisy wheel.

**Program.** As a noun: a set of instructions given to a computer to enable it to perform a particular function. As a verb: to input such information into a computer.

**Programming language.** A language used in the composition of a computer program.

**PROM.** Programmable Read Only Memory. A permanent storage system for data that can be programmed both by the manufacturer and by the user.

**Quad-density.** Refers to a two-sided, double-density disk which is able to store four times the information of a one-sided, normal density disk.

**RAM.** Random Access Memory. Volatile storage system for data that can be changed—added to, subtracted from, rearranged—by the user. Information stored in RAM must be saved before the system is shut off or it will be lost.

**Read/write.** Signifies that information can be both read from and written into memory (RAM or permanent storage).

**Resolution.** A measure of the sharpness of a CRT (cathode ray tube) picture. Expressed in pixels, it can refer to either the number of scanning lines on the terminal or the number of addressable pixels on the screen.

**Reverse video.** The displaying of dark characters on a light background.

**RF Modulator.** Used to change computer video signals into radio frequency signals that can be picked up by a TV antenna and displayed.

**RGB.** Video signal composed of red, green, and blue that has much higher resolution and brighter colors than the standard composite color TV signals. A monitor with three separate electron guns, rather than the single gun used by the average color TV, is required.

**ROM.** Read Only Memory. Data that is built into a computer or software, and cannot be changed, such as the information that operates the computer immediately after start-up. In a computergame, for example, backgrounds which do not change or are not affected by gameplay are ROM. Spaceships and projectiles, which constantly are updated, are RAM.

**Save.** Recording information from the computer's memory onto tape or a disk.

**Scrolling.** Moving up and down (or from left to right) through the displayed information contained in the computer's memory, only a certain portion of which can be displayed on the terminal screen at one time.

**Sector.** A part of the track of a storage disk.

**Serial.** An input or output system that submits one bit of data at a time.

**Serial port.** A computer's input or output port through which data is transmitted in serial fashion. This is generally done, with home computers, through an RS232C serial interface port.

**Single-sided.** A disk with only one side that can store data.

**Software.** Programs to run on a computer on tape, disk, or cartridge. Everything from *Donkey Kong* to *Data Base*.

**Source code.** Program written in English-like words via an editor program which needs to be translated (with an assembler or compiler) into a language the computer understands.

**Sprite.** A high resolution programmable object. Any graphic figure is a sprite; sprites can be changed and animated through commands in BASIC

**Syntax.** Rules of grammatical usage governing programming language, as with English and other languages.

**Terminal.** The screen that displays computer information.

**Text.** The words and numbers displayed on the CRT screen.

**Tractor feed.** Mechanism that holds fan-fold paper in place and moves it through the printer, using sprockets that fit into the holes on both sides of the paper.

**Volatile storage.** A storage device, such as RAM, that loses the data it contains when power is cut off.

**Winchester.** A variety of hard disk that is sealed in a container.

**Word processor.** Can refer to the complete system, usually consisting of computer, display, memory storage, and printer, that is used to produce written documents; also refers to a software program that enables a computer to perform word processing functions.

**Write.** The process of transferring data from memory to permanent storage.

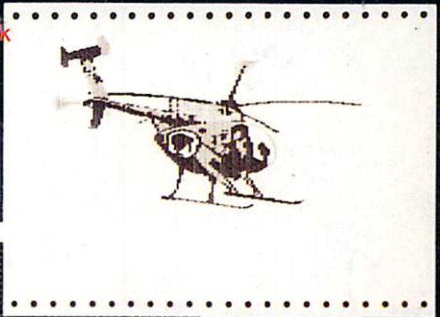
**Write protection.** Process that prevents writing to, or erasing from, a disk. With 8-inch disks, this is accomplished by removing a tab from the jacket; With 5¼ disks, by affixing the tab to the jacket.

# CADPIC™

## CADPIC IS THE TOTAL COMPUTER-AIDED DESIGN PACKAGE FOR THE C-64™

- **EDUCATION** — LEARN PAINTING, GRAPHIC DESIGN AND DRAFTING IN FULL COLOR
- **FUN** — DRAW & PRINT CARTOONS, COMIC STRIPS, COLORING-IN & MICRON EYE™ CAMERA PICTURES
- **BUSINESS** — GRAPHICS OF EVERY KIND, SHADED PRINTOUT, COMPUTER COLOR SEPARATIONS, ADS
- **HOBBY** — TAPESTRY, EMBROIDERY, HOOKED RUG DESIGNS, AND, OF COURSE, 16 COLOR PAINTING

PAINTPIC™ available on Diskette for $39.95
Create 16 color paintings / Pens & brushes for special effects / Draw with CRSR or Joystick / Automatic shapes (filled & unfilled) / Perspective & Home points / Copying / Mirroring / Rotation / Halving / Doubling / Load & save pictures or blocks / Lots of help screens.





"VENUS" BY VELÁZQUEZ, A PAINTPIC IMAGE



+

PRINTAPIC™ Available on Diskette for $44.95
Print Paintpic pictures in advanced grey shades or high-resolution black & white / Tapestry, Embroidery, Hooked Rug, 'Paint by number' prints / Convert and print Micron™ images / Color separations for publication, Ads, business reports / Works for most printers / Includes 'Venus' above, and helicopter.

## TOGETHER AS CADPIC, FOR ONLY $79.95 – *THERE'S NOTHING LIKE IT!!*

**GIANT COMPUTER ART PRINTS**
**NOW YOU CAN GET "ROOM", "STORM" AND "HOLYLAND" IN 20"x 24" POSTERS**




**$8.95 EACH + $1.00 POSTAGE AND HANDLING OR, SET OF 3 FOR $25 POSTAGE FREE**

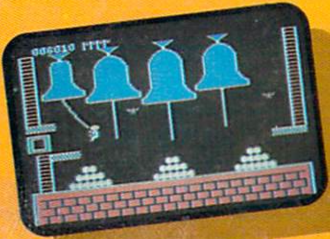
CADPIC™ comes with complete instructions. Payment in advance in U.S. Dollars by check or money order or via VISA, AMEX. Add $2.00 postage and handling. California residents add 6% sales tax. Paintpic available on cassette $35.

**Dealer Enquiries Welcome.** **Product Brochure.** **Phone 714-261-5114**

**KIWISOFT™ PROGRAMS**
18003-LSkypark South, Irvine, CA92714

MicronEye is a trademark of Micron Technology, Inc. Commodore 64 is a trademark of Commodore Electronics, Ltd.

# ZAXXON FOR THE C-64!








**Quasimodo**
Quasimodo knows who stole the crown jewels. He even knows where they are, but the soldiers just won't leave him alone! This multi-screen arcade adventure is a great combination of skill and strategy.
Commodore 64 disk & cassette




**ZAXXON**
At last, your favorite arcade game comes home to the C-64! Amazing 3-D graphics, a diagonally scrolling screen and incredible sound effects guarantee hours of space excitement. And if you practice hard, you might be able to meet the deadly ZAXXON himself.
Commodore 64 disk & cassette




ZAXXON, a trademark of SEGA Enterprises, Inc.

**Slamball**
Maybe you've played pinball before, but not like this! Up and down scrolling over 4 full screens, plus complete ball control and even a "tilt" feature in the best computer pinball game yet.
Commodore 64 disk & cassette




All Synapse entertainment titles are available at software dealers everywhere, or direct from Synapse for $34.95 ($39.95 for ZAXXON) plus $2 handling. Send check, money order or VISA/MasterCard number.

**More C-64 Titles:**
Blue Max, Fort Apocalypse, Shamus and Shamus Case II, Necromancer, Pharaoh's Curse, Zeppelin, Drelbs, Sentinel.

Write for a FREE Synapse catalog. Synapse games are also available for the Atari, Apple and IBM computers.

Name__________
Address__________
City__________
State__________ Zip__________
Computer__________ C

**synapse**
5221 Central Avenue, Richmond, CA 94804 · 415/527-7751



Reader Service No. 116